

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

**THIS PAGE BLANK (USPTO)**

19 BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENTAMT

Offenlegungsschrift  
DE 3837 590 A 1

- 21 Aktenzeichen: P 38 37 590.7  
22 Anmeldetag: 5. 11. 88  
43 Offenlegungstag: 10. 5. 90

51 Int. Cl. 5:  
H 04 N 7/13  
H 04 N 7/12

DE 3837 590 A 1

71 Anmelder:  
ANT Nachrichtentechnik GmbH, 7150 Backnang, DE

72 Erfinder:  
Gilge, Michael, Dipl.-Ing., 5100 Aachen, DE

Prüfungsantrag gem. § 44 PatG ist gestellt

54 Verfahren zum Reduzieren der Datenrate von digitalen Bilddaten

Zum Reduzieren der Datenrate von digitalen Bilddaten werden für Teilbildbereiche in zeitlich aufeinanderfolgenden Bildern Bewegungsvektoren dadurch ermittelt, daß bezüglich zweier Bilder jeweils die Abweichung von der Position des unverschobenen Teilbildbereichs zur tatsächlichen Position des Teilbildbereichs im zeitlich folgenden Bild ermittelt wird.

Bei der Erfindung werden alle möglichen Bewegungsvektoren innerhalb eines Teilsuchgebietes durch nur eine Ersatzverschiebungsposition charakterisiert. Ausgewählt und anschließend für die Übertragung codiert wird jene Ersatzverschiebungsposition, bei der die Ähnlichkeit der Teilbildbereiche zwischen zwei zeitlich aufeinanderfolgenden Bildern am größten ist.

Aus den ermittelten Bewegungsvektoren wird ein Schätzbild erstellt und durch Differenzbildung mit dem Originalbild ein Prädiktionsfehlerbild gebildet. Die zu übertragende Fehlermenge aus dem Prädiktionsbild wird anhand eines Fehlerkriteriums ausgewählt. Die Signalisierung der ausgewählten Bildteile wird über einen »Quadtree« codiert. Das Prädiktionsfehlerbild wird klassifiziert und der Prädiktionsfehler entsprechend der Klassifizierung adaptiv grobquantisiert. Zur Grobquantisierung werden Strukturblöcke gebildet. Für Bildbereiche, in denen eine Bewegungskompensation versagt, wird Originalbildinformation in dekorrelierter Form übertragen.

Das Verfahren nach der Erfindung eignet sich insbesondere zur Bildübertragung für einen 64 kbit/s-Kanal im ISDN-Netz.

DE 3837 590 A 1

Die Erfindung betrifft ein Verfahren gemäß dem Oberbegriff des Patentanspruchs 1.

## 1. Stand der Technik

Aus EP 236 519 A1 ist es bekannt, Bewegungsvektoren für Teilbildbereiche in zeitlich aufeinanderfolgenden Bildern einer Bildsequenz zu ermitteln, indem bezüglich zweier Bilder jeweils die Abweichung von der Position des unverschobenen Teilbildbereichs zur tatsächlichen Position des Teilbildbereichs in einem zeitlich folgenden Bild ermittelt wird. Die Bewegungsvektoren werden mittels eines Iterationsprozesses mit jedem Schritt genauer bestimmt.

Aus der DE 37 04 777 C1 geht hervor, eine DPCM-Schleife mit Quantisierung in Zusammenhang mit einer Prädiktionsfehlercodierung zur Bilddatenreduktion zu verwenden. Für stark bewegte Bildbereiche werden bildpunktabhängige Informationen, die beispielsweise einer DCT-Codierung unterzogen werden, übertragen.

Zur Prädiktion wird im einfachsten Fall der vorhergehende "Frame" verwendet und das Bild in Blöcke konstanter Größe segmentiert. Ein Bildblock wird nur dann codiert und übertragen, wenn der Prädiktionsfehler eine vorgegebene Schwelle überschreitet (Conditional Replenishment) [1].

Durch Algorithmen zur Bewegungsschätzung kann das Verfahren verbessert werden. Dazu werden Verschiebungsvektoren ermittelt, welche entweder einzelnen Bildpunkten, Bildblöcken oder Objekten zugeordnet sind. Bei punktwiser Bewegungsschätzung wird der Bewegungsvektor meist rekursiv aus zwei Bildern einer Bildsequenz bestimmt (pel recursive) [2], [3], [4]. Zur blockweisen Schätzung wird der zu übertragende Block innerhalb eines begrenzten Gebietes des vorherigen Bildes verschoben und für jede Position ein Fehlermaß berechnet. Die Position mit minimalem Fehler ergibt den Verschiebungsvektor für den Block (blockmatching). In der Literatur wird als Fehlerkriterium neben dem mittleren quadratischen Fehler (MSE) hauptsächlich die mittlere absolute Differenz (MAD) verwendet. Zur Beschleunigung der Suche nach dem minimalen Fehler werden meist schnelle Algorithmen eingesetzt [5], [6], [7]. Eine Erweiterung des "blockmatching" ist dabei das sogenannte "objectmatching" [8]. Dazu wird das Bild in bewegte Objekte segmentiert und die Bewegungsinformation dem kompletten Objekt zugeordnet.

Zur Erzielung extrem niedriger Datenraten werden praktisch ausschließlich die beiden letztgenannten Verfahren verwendet, da die hier zu übertragenden Zusatzinformationen für die Verschiebungsvektoren geringer ist. Im Falle des "objectmatchings" steht dem Vorteil nur einen Vektor pro Objekt übertragen zu müssen der Nachteil gegenüber, daß auch die Form der Objekte beschrieben werden muß. In [8] wird dafür eine relativ grobe Block-Approximation verwendet.

Grundsätzlich wird bei allen Bewegungsschätzverfahren zwischen Pel-Genauigkeit (Integer-Genauigkeit) und Sub-Pel-Genauigkeit unterschieden. Pel-Genauigkeit bedeutet, daß sowohl die Bewegung im Bild beliebig fein sein kann, die Bewegungsvektoren nicht genauer als auf einen Bildpunktstand genau berechnet werden. Dieser gröberen Bewegungsschätzung steht der Vorteil gegenüber, daß keine rechenaufwendige Interpolation zur Erreichung der Sub-Pel-Genauigkeit durchgeführt werden muß.

Die Bewegungsvektoren werden anschließend auf das vorherige Bild angewendet, um durch Verschiebung von Bildteilen des alten Bildes eine Schätzung des neuen Bildes zu erhalten (Bewegungskompensation) [9], [10]. Die Grenzen des mit Bewegungskompensation zu erzielenden Codiergewinns bei hybriden Verfahren werden in [11] auf Basis der Rate-Distortion Theorie untersucht. Die dargestellten Ergebnisse gelten allerdings nur für hohe S/N-Verhältnisse und sind auf die Codierung für eine Rate von 64 kbit/s wahrscheinlich nur bedingt anwendbar.

Nach Subtraktion des bewegungskompensierten Schätzbildes vom aktuellen Bild verbleibt das Bild des Prädiktionsfehlers (Abb. 4.4.1a). Zur Übertragung dieses Fehlerbildes wird meist eine Blocksegmentierung in Blöcke der Größe  $16 \times 16$  oder  $8 \times 8$  durchgeführt und diese mit den üblichen Verfahren der Standbildcodierung, wie zum Beispiel Vektorquantisierung [12] oder diskrete Cosinus Transformation mit Quantisierung im Spektralbereich [13], codiert übertragen.

Um zu einer konstanten Datenrate zu gelangen wird meist ein Puffer verwendet, dessen Füllstand die Quantisierung oder andere Coderparameter steuert [14]. Bei großem Bewegungsanteil im Bild werden aufgrund des Ausgleichs durch grobere Quantisierung und der blockweisen Verarbeitung sehr schnell Block-Strukturen sichtbar (blocking-effect). Andere störende Effekte sind der sogenannte "dirty-window"-Effekt: Durch die grobe Quantisierung im Spektralbereich werden teilweise die Basisbilder der verwendeten Transformation sichtbar, die durch die starre Blocksegmentierung im Gegensatz zum Bildinhalt stillstehen. Dadurch hat der Betrachter den Eindruck, die Szene durch ein verschmutztes Fenster zu beobachten. Ein weiterer störender Effekt ist der sogenannte "mosquito"-Effekt, der ebenfalls durch zu grobe Quantisierung hervorgerufen wird und an der Grenze zwischen bewegtem Objekt und stillstehendem Hintergrund einen flirrenden Bildeindruck erzeugt. Die bewegten Bildteile erscheinen wie von Moskitos "umschwirrt". Der überaus störende Bildeindruck durch Sichtbarwerden des Blockrasters bei zu hohem Bewegungsanteil (blocking-effect) wird zusätzlich dadurch verstärkt, daß das verwendete Blockraster von Blöcken der Größe  $8 \times 8$  oder  $16 \times 16$  zu Ortsfrequenzen führt, die gerade im Bereich maximaler Sensivität des menschlichen Sehapparates liegen [15].

Da das vorherrschende Bildmaterial meist keine Auflösung von Bewegungen mit 25 Hz erfordert, wird oft eine zeitliche Unterabtastung vorgenommen. Dazu wird nur jeder zweite oder jeder dritte Frame übertragen. Die pro Bild zur Verfügung stehende Datenrate verdoppelt oder verdreifacht sich also. Durch eine Filterung der Eingangssequenz mit Hilfe eines meist rekursiven Tiefpaßfilters kann die Effizienz der Prädiktion durch stärker werdende zeitliche Korrelation gesteigert werden. Auf der Empfängerseite ist eine Rückgewinnung der ausgelassenen Frames erforderlich. Im einfachsten Fall wird ein Frame wiederholt, also mehrfach gezeigt (frame

repetition). Dies führt jedoch zu einem ruckhaften Ablauf der Bewegungen. Ein anderes Verfahren erzeugt die ausgelassenen Frames durch lineare Interpolation zwischen den übertragenen Frames. Die Bewegungen erscheinen hierbei zwar flüssiger, doch es entstehen störende Doppelkanten an bewegten Luminanzsprüngen. Dies kann nur durch eine bewegungsadaptierte Interpolation vermieden werden. Dazu ist eine Segmentierung des Bildes in bewegte Objekte, freiwerdenden und bedeckten Hintergrund nötig. Außerdem sind Bewegungsvektoren erforderlich, die mit der realen Bewegung der Objekte übereinstimmen. Die zur Erzielung eines minimalen Prädiktionsfehlers ermittelten Vektoren, insbesondere wenn schnelle Suchalgorithmen verwendet werden, sind hierzu ungeeignet.

## 2. Aufgabe und Lösung

Aufgabe der Erfindung ist es, ein Verfahren ausgehend vom Oberbegriff des Patentanspruchs 1 anzugeben, welches zu einer effektiven Bildcodierung, insbesondere des Prädiktionsfehlers, bei einer vorgegebenen niedrigen Datenrate, z. B. 64 kbit/s, führt, wobei Bewegungsabläufe unter Berücksichtigung des menschlichen visuellen Systems möglichst natürlich erscheinen sollen.

Diese Aufgabe wird durch die kennzeichnenden Merkmale des Patentanspruchs 1 gelöst. Die Unteransprüche zeigen vorteilhafte Weiterbildungen dieses Verfahrens auf.

Das Verfahren nach der Erfindung gewährleistet eine zuverlässige Bildrekonstruktion auch für schnelle Suchalgorithmen. Der Hardwareaufwand für die Aufbereitung des Übertragungssignals ist geringer als bei den meisten bekannten Verfahren zur Bilddatenreduktion.

Anhand der Abbildungen und Tabellen werden Ausführungsbeispiele der Erfindung sowie bisherige Lösungen der Bilddatenreduktion im Zusammenhang mit der Erfindung nun näher erläutert. Es zeigen:

**Abb. 3.1.1:** Suchschritte bei 2D Logarithmic Search für einen Verschiebungsvektor (2, 6), Anzahl der Schritte 5, nach [21];

**Abb. 3.1.2:** Suchschritte bei Three step Verfahren und Verschiebungsvektor (2, 6) nach [21];

**Abb. 3.1.3:** Suchschritte bei Conjugate Direction Search für einen Verschiebungsvektor (2, 6), Anzahl der Schritte: 9, nach [21];

**Abb. 3.2.1:** Verlauf des Fehlerkriteriums über Verschiebung im gesamten Suchbereich;

**Abb. 3.2.2:** Suchpositionen für modifizierte volle Suche, d. h. volle Suche mit nachgeschaltetem Vektorquantisierer und synthetischem Codebuch, Bonus für Nullvektor zur Segmentierung in bewegte/unbewegte Bildteile;

**Abb. 3.2.3:** Verteilungsdichtefunktion der Vektorzustände;

**Abb. 3.2.4:** Bewegungsvektorfeld ohne Nullvektorbonus;

**Abb. 3.2.5:** Bewegungsvektorfeld mit Nullvektorbonus;

**Abb. 3.2.6:** Leistungsvergleich der schnellen Suchverfahren, alle Verfahren arbeiten mit Segmentierung, Pel-Genauigkeit und Suchbereich  $\pm 6$ ;

**Abb. 3.2.7:** Vergleich der modifizierten vollen Suche mit voller Suche und 3-step Verfahren, Bedingungen wie in Abb. 3.2.6;

**Abb. 3.2.8:** Einfluß der Größe des Suchgebietes, volle Suche mit  $\pm 6$  bzw.  $\pm 11$  pels Suchbereich;

**Abb. 3.2.9:** Prinzip der hierarchischen Suche am Beispiel eines Bewegungsvektors:

1. Stufe: modifizierte volle Suche mit 1 bzw. 2 pel Auflösung;

2. Stufe: volle Suche mit eingeschränktem Suchbereich und 0,5 pel Auflösung;

**Abb. 3.2.10:** Vergleich der hierarchischen Suche (0,5 Pel Genauigkeit) mit voller Suche (1 Pel) und modifizierter voller Suche (1 Pel);

**Abb. 3.2.11:** Simulation der hierarchischen Bewegungskompensation:

oben links: Originalbild N

oben rechts: Originalbild N + 3

unten links: Differenzbild ohne Bewegungskompensation (2fach verstärkt)

unten rechts: Differenzbild mit Bewegungskompensation (2fach verstärkt);

**Abb. 3.2.12:** Differenzbilder (2fach verstärkt) bei verschiedenen Suchalgorithmen:

oben links: 3-step-Verfahren

oben rechts: volle Suche

unten links: modifizierte volle Suche (mit Vektorquantisierung) Auflösung 1 pel, Suchbereich  $\pm 11$  pel

unten rechts: hierarchische Suche, Auflösung: 0,5 pel Suchbereich:  $\pm 11$  pel;

**Tab. 3.3.1:** Korrelationskoeffizienten örtlich benachbarter Bewegungsvektoren;

**Tab. 3.3.2:** Entropie der Bewegungsvektoren;

**Abb. 3.3.3:** Prinzip der Quadtree-Codierung;

**Tab. 3.3.4:** Quadtree-Codierung der Bewegungsvektoren;

**Abb. 4.1.1:** Autokorrelationsfunktion in horizontaler (—) und in vertikaler (---) Richtung, Originalsequenz: "Miß Amerika";

**Abb. 4.1.2:** 2-dimensionale Autokorrelationsfunktion für die Originalsequenz "Miß Amerika";

**Abb. 4.1.3:** Autokorrelationsfunktion in horizontaler (—) und in vertikaler (---) Richtung, Sequenz der Prädiktionsfehlerbilder;

**Abb. 4.1.4:** 2-dimensionale Autokorrelationsfunktion für die Sequenz der Prädiktionsfehlerbilder;

**Abb. 4.1.5:** Schaltung zur Gewinnung der Hochpaßkomponente mittels Tiefpaßfilterung des Bildes;

**Abb. 4.1.6:** Modellierung der Statistik des Prädiktionsfehlersignals durch ein hochpaßgefiltertes Bildsignal, Vergleich der Autokorrelationsfunktionen:

— AKF Prädiktionsfehlerbilder

--- AKF Hochpaßkomponenten der Originalbilder, Filterung: 7 • 7 FIR Tiefpaß;

Abb. 4.1.7: Autokorrelationsfunktion eines ideal hochpaßgefilterten Markov-Prozesses erster Ordnung nach Gl. 4.1.21:  $k = 2 \cdot \pi \cdot f_0 \cdot \tau$ ,  $f_0 \approx 1$  MHz;

Abb. 4.2.1: Eigenwerte der Kovarianzmatrixen für Originalbilder (—) und für Prädiktionsfehlerbilder (---) normiert auf den größten Eigenwert  $E(1)$ ;

Abb. 4.2.2: Varianten der DCT-Spektralkoeffizienten (Blockgröße  $8 \times 8$  Bildpunkte) für Originalbilder (—) und für Prädiktionsfehlerbilder (---), sortiert nach dem Betrag der Varianz;

Abb. 4.3.2: Übertragene Bildsignalleistung in Abhängigkeit der Anzahl übertragener DCT-Koeffizienten bei Anwendung der DCT auf Originalbilder (—) und Prädiktionsfehlerbilder (---);

Abb. 4.2.4: Verteilungsdichtefunktion des Energiekompressionskoeffizienten für Originalbilder (—) und für Prädiktionsfehlerbilder (---);

Abb. 4.3.1: Prädiktionsfehlerbild;

Abb. 4.3.2: Verteilungsdichtefunktion des Prädiktionsfehlerbildes;

Abb. 4.3.3: Modulationsübertragungsfunktion des visuellen Systems (nach Campbell und Robson) [26] 1;

Abb. 4.3.4: Blockschaltbild der Strukturcodierung;

Abb. 4.3.5: Verteilungsdichtefunktion des Prädiktionsfehlerbildes nach Klassifikation in 4 Klassen;

Abb. 4.3.6: Auftrittshäufigkeit der 81 möglichen Blocktypen;

Abb. 4.3.7: Auftrittshäufigkeit der verbleibenden 31 Blocktypen;

Abb. 4.4.1: Blockschaltbild des Coders (a) herkömmlich, (b) erfindungsgemäß;

Abb. 4.4.2: Quadtree-Maskenbild zur Codierung der Orts- und Klasseninformation;

Abb. 4.4.3: Codierte Prädiktionsfehlerbild;

Abb. 4.4.4: Rekonstruierte Bilder der Sequenzen "MISS AMERIKA" und "ALEXIS" nach Codierung mit 64 kbit/s.

Bevor auf das Verfahren nach der Erfindung im einzelnen eingegangen wird, seien zunächst einige Grundbegriffe und bekannte Verfahren erläutert.

### 3. Bewegungsschätzung und Kompensation

Die Verarbeitung von Bildsequenzen ist für sehr viele Anwendungsgebiete [16] von Bedeutung, von denen einige hier aufgezählt sind:

- \*) Industrie: Prozeßüberwachung, "robot vision"
- \*) Medizin: Bewegungs Studien aus Röntgenfilmszenen oder Zellbewegungen
- \*) Meteorologie: Verfolgung von Wolkenbewegungen
- \*) Verkehrstechnik: Überwachung des Straßenverkehrs
- \*) Nachrichtentechnik: Bilddatenkompression für Bildtelefon, Videokonferenzen oder HDTV-Übertragung

In allen Fällen ist die Bewegungsanalyse von großer Bedeutung. Bewegungen in der Szene äußern sich in Änderungen des Helligkeitsverlaufes von Bild zu Bild. Diese Änderungen können zur Schätzung der Bewegung herangezogen werden. Für den Anwendungsfall der Bewegtbildcodierung, der hier interessieren soll, kann die Effizienz von DPCM-Verfahren durch Bewegungsschätzung und Kompensation erheblich gesteigert werden. Dazu wird die Bewegung zwischen zwei oder mehr aufeinanderfolgenden Bildern geschätzt und zur Verbesserung der Prädiktion eingesetzt. Die errechnete Bewegungsinformation in Form eines Bewegungsvektorfeldes unterscheidet sich je nach eingesetztem Verfahren hinsichtlich des zugrundeliegenden Bewegungsmodells und der Genauigkeit der Vektoren:

Die meisten zur Beschreibung der Bewegung herangezogenen Modelle sind aus Gründen des Aufwandes recht einfach gehalten und gehen zum Beispiel von starrer translatorischer Bewegung der Objekte in der Szene aus. Andere Bewegungsformen, wie z. B. Rotation oder Verformung der Objekte, müssen durch Verschiebung starrer Bildteile approximiert werden. Dies ist nur bei kleinen Bewegungen hinreichend exakt.

Bei der Genauigkeit der Vektoren unterscheidet man zwischen pel- und sub-pel-Genauigkeit:

Im ersten Fall werden die Vektoren nur auf einen Bildpunkt genau (Integer) berechnet, obwohl die Bewegung in der Szene natürlich beliebig fein erfolgen kann. Zur Erzielung größerer Genauigkeit (sub-pel) ist eine Interpolation der Bildpunkte erforderlich, was einen erhöhten Aufwand bedeutet.

Soll die Bewegungsinformation auch zur bewegungsrichtigen Interpolation ausgelassener Bilder im Empfänger eingesetzt werden, so ist die Zuverlässigkeit der Vektoren zur Beschreibung der wahren Bewegung unabdingbar. Während für die Prädiktion Mehrdeutigkeiten der Vektoren, d. h. verschiedene Vektoren führen zum gleichen Schätzbild, nicht negativ in Erscheinung treten, ist dies bei bewegungskompensierter frame-Interpolation oder frame-Extrapolation nicht der Fall. In diesem Fall muß auch eine Überschreitung des Vektor-Wertegebietes vermieden werden.

Im folgenden werden zunächst verschiedene, aus der Literatur bekannte, Verfahren zur Bewegungsschätzung vorgestellt. Die Algorithmen besitzen entweder eine hohe Komplexität die Hardwareimplementierungen nicht zugänglich ist, oder postulieren Modelle, die der Praxis oft nicht hinreichend entsprechen. Zur Lösung dieser Probleme wird ein neues zweistufiges Verfahren vorgestellt und in Bezug auf Leistungsfähigkeit und Aufwand mit anderen Ansätzen verglichen. Schließlich wird eine geeignete Codierung der Bewegungsvektordinformation diskutiert.

## 3.1 Algorithmen zur Bewegungsschätzung

## 3.1.1 Verfahren im Spektralbereich

Nach dem Verschiebungssatz der Fourier-Transformation kann eine Verschiebung im Originalbereich aus der Phasendifferenz zwischen den Fourier-Transformierten des Originals und des verschobenen Bildes berechnet werden:

$$F(u,v) | f(x,y) \quad (3.1.1)$$

$$G(u,v) | g(x,y) = f(x-\Delta x, y-\Delta y) \quad (3.1.2)$$

$$\Delta \varphi(u,v) = \arg(G(u,v)) - \arg(F(u,v)) = -2\pi \cdot (u \cdot \Delta x + v \cdot \Delta y) \quad (3.1.3)$$

Bestimmt man  $\Delta \varphi$  für zwei Frequenzpaare  $(u,v)$  kann daraus  $(\Delta x, \Delta y)$  berechnet werden. Zu beachten ist allerdings, daß die Berechnung der Phasenverschiebung wegen der Mehrdeutigkeit der Arkustangensfunktion nicht eindeutig ist. Dieses Problem kann jedoch unter Berücksichtigung des Bereiches der möglichen Verschiebung gelöst werden. Schwerwiegender ist die Forderung, daß das Objekt sich vor konstantem Hintergrund bewegen muß. Andernfalls erhält man  $g(x,y)$  nicht durch Verschiebung von  $f(x,y)$  wie in Gleichung 3.1.3 (freierwerdender bzw. verdeckter Hintergrund). Eine Lösung für den Fall nicht-uniformen Hintergrunds ist die Segmentierung der bewegten Objekte zur Trennung vom Hintergrund oder der Einsatz der Phasenkorrelationstechnik. Letzteres Verfahren ist schon vor einiger Zeit [17] veröffentlicht worden und ist wie folgt definiert [18]:

$$PCF(x,y) = F^{-1} \left( \frac{F(g(x,y))^* \cdot F(f(x,y))}{|F(g(x,y))^* \cdot F(f(x,y))|} \right) \quad (3.1.4)$$

Dabei bezeichnen  $F, F^{-1}$  die Fourier-, bzw. inverse Fourier-Transformation und das Symbol  $*$  konjugiert komplex. Die Korrelation wird über einen Bildblock durchgeführt, dessen Dimensionen vorzugsweise Zweierpotenzen sein sollten, um den Einsatz der FFT zu ermöglichen. Die Stelle des Maximums der Phasenkorrelationsfunktion  $PCF(x,y)$  korrespondiert zu der Verschiebung des Gebietes. Wegen des erforderlichen Rechenaufwandes sollen die Verfahren im Frequenzbereich zur Anwendung in einem Bildtelefon hier nicht weiter in Betracht gezogen werden.

## 3.1.2 Differentielle Verfahren

Unter der Voraussetzung, daß sich nur die Positionen der Bildpunkte, nicht aber deren Helligkeiten ändern, lassen sich zeitliche Differenzen in örtliche Differenzen überführen. Die Helligkeitsänderung läßt sich bei reiner Translation durch die Gleichung:

$$f(x,y,t) = f(x-dx, y-dy, t-1) \quad (3.1.5)$$

beschreiben. Die Differenz zeitlich aufeinanderfolgender Frames beträgt:

$$\Delta f(x,y) = f(x,y,t) - f(x,y,t-1) \quad (3.1.6)$$

Unter den oben getroffenen Bedingungen kann die zeitliche Veränderung des Luminanzsignals in eine örtliche Veränderung überführt werden:

$$\Delta f(x,y) = f(x,y,t) - f(x+\Delta x, y+\Delta y, t) \quad (3.1.7)$$

Entwickelt man den zweiten Term in eine Taylorreihe, die unter der Voraussetzung kleiner Bewegungen nach dem ersten Glied abgebrochen werden kann, erhält man:

$$f(x+\Delta x, y+\Delta y) = f(x,y) + \frac{\partial f(x,y)}{\partial x} \cdot \Delta x + \frac{\partial f(x,y)}{\partial y} \cdot \Delta y + \dots \quad (3.1.8)$$

$$\Delta f(x,y) = - \frac{\partial f(x,y)}{\partial x} \cdot \Delta x - \frac{\partial f(x,y)}{\partial y} \cdot \Delta y \quad (3.1.9)$$

Für jeden Bildpunkt  $(x,y)$  können nun die Ableitungen und der Wert  $\Delta f(x,y)$  berechnet werden. Durch Mittelwertbildung über das bewegte Gebiet erhält man eine Schätzung für  $(\Delta x, \Delta y)$  [19]. Da der beschriebene Algorithmus einen linearen Verlauf der Helligkeit am Punkt der Ableitung voraussetzt, können nur kleine Bewegungen hinreichend genau bestimmt werden. Durch rekursive Verfahren [2], 20 können einerseits größere Verschiebung berechnet werden, andererseits ergibt sich eine verbesserte Genauigkeit der Vektoren. Dazu wird, ausgehend von einer Schätzung für den Bewegungsvektor  $BW^i$ , ein verbesserter Vektor  $BW^{i+1}$  berechnet:

$$BW^{l+1} = BW^l + U^l \quad (3.1.10)$$

Der "Update"-Term  $U^l$  zur Verbesserung des Bewegungsvektors  $BW^l$  ergibt sich aus der Minimierung der sogenannten "displaced frame difference" (DFD). Diese errechnet sich unter Verwendung des bisherigen Bewegungsvektors  $BW^l$  wie folgt:

$$DFD(x, y, BW^l) = f(x, y, t) - f(x - \Delta x^l, y - \Delta y^l, t-1) \quad (3.1.11)$$

Die verschiedenen Verfahren unterscheiden sich durch die Berechnung des Minimums von DFD. Der Genauigkeit der Verfahren steht ein erheblicher Rechenaufwand entgegen. Zusätzlich entstehen infolge der rekursiven Struktur nicht-konstante Coderdurchlaufzeiten, die die Eignung der Verfahren für eine Hardwarerealisierung in einem Bildtelefon in Frage stellen.

### 3.1.3 Suchverfahren ("matching"-Verfahren)

Anstelle einer rekursiven Bestimmung des Bewegungsvektors kann eine Kreuzkorrelation der zu betrachtenden Gebiete aufeinanderfolgender Frames vorgenommen werden. Der Verschiebungsvektor ergibt sich aus der Position der maximalen Korrelation. Je nach Form der miteinander korrelierten Gebiete unterscheidet man das sogenannte "object matching" oder "block matching" Verfahren.

Objektorientierte Verfahren zur Bewegungsschätzung (object matching) [8], [21] zerteilen das Bild in Bereiche die zu den Objekten in der Szene korrespondieren und für die dann jeweils ein Bewegungsvektor ermittelt wird. Es ist also eine Segmentierung des Bildes in zusammenhängend bewegte Bildteile erforderlich. Dies ist einerseits mit hohem Rechenaufwand verbunden, andererseits muß dem Empfänger neben der eigentlichen Bewegungsinformation die Form der Objekte mitgeteilt werden. Zur Beschreibung der Bewegung der Einzelobjekte sollten neben reinen translatorischen Verschiebungen auch Rotationen und ggf. Größenänderungen (Zoom) berücksichtigt werden. In [8] werden die Objekte durch Blöcke der Größe  $8 \cdot 8$  approximiert, indem man zusammenhängende Blöcke mit gleichem Bewegungsvektor einem Objekt zuordnet.

Beim block matching wird das Bild in rechteckige Blöcke der Größe  $M \cdot N$  Bildpunkte segmentiert. Dabei werden meist Blöcke der Größe  $8 \cdot 8$  oder  $16 \cdot 16$  verwendet. Unter der Annahme, daß alle Bildpunkte innerhalb der Blöcke der gleichen Bewegung unterliegen, werden die Blöcke innerhalb eines Suchbereiches verschoben und mit dem vorhergehenden Frame korreliert. Es wird also nur ein Bewegungsvektor pro Block bestimmt. Rotationen und Größenänderungen von Bildobjekten können bei genügend kleiner Blockgröße und kleiner Bewegung durch unterschiedliche Translationen der einzelnen Blöcke ausreichend genau beschrieben werden. Aus der maximalen Verschiebung  $D_{max}$  in horizontale und vertikale Richtung und einer Blockgröße von  $N \cdot M$  Punkten errechnet sich die Fläche des Suchgebietes:

$$AR = (M + 2 \cdot D_{max}) \cdot (N + 2 \cdot D_{max}) \quad (3.1.12)$$

Zum Auffinden des Minimums der Kreuzkorrelationsfunktion wird im allgemeinen ein Fehlerkriterium  $E$  definiert und nach der Position des Minimums von  $E$  gesucht. Häufig verwendete Fehlerkriterien sind der mittlere quadratische Fehler MQF oder die mittlere absolute Differenz MAD:

$$MQF(\Delta x, \Delta y) = \frac{1}{M \cdot N} \cdot \sum_{m=1}^M \sum_{n=1}^N (f(m, n, t) - f(m + \Delta x, n + \Delta y, t-1))^2 \quad (3.1.13)$$

$$MAD(\Delta x, \Delta y) = \frac{1}{M \cdot N} \cdot \sum_{m=1}^M \sum_{n=1}^N |f(m, n, t) - f(m + \Delta x, n + \Delta y, t-1)| \quad (3.1.14)$$

Die Bestimmung des MOF Fehlermaßes erfordert für jede verglichene Vektorposition je  $N \cdot M$  Subtraktionen, Multiplikationen und Additionen. Bei einem Suchfenster von  $D_{max} = 6$  Bildpunkten in jeder Richtung sind zur Bestimmung des Bewegungsvektors für einen Block bei voller Suche

$$Q_{fs} = (2 \cdot D_{max} + 1)^2 = 13 \cdot 13 = 169 \text{ (full search)} \quad (3.1.15)$$

dieser Berechnungen notwendig. Zur Verringerung dieses Rechenaufwandes werden in der Literatur verschiedene schnelle Algorithmen angegeben [21]:

Das "2D-logarithmic" Suchverfahren [5] ist schematisch in Abb. 3.1.1 dargestellt. In jeden Suchschritt wird jeweils eine vierer Nachbarschaft auf minimalen Fehler untersucht. Die Suche wird vom so gefundenen Punkt minimalen Fehlers solange fortgesetzt; bis keine Verbesserung des Fehlers mehr erzielt werden kann oder der Rand des Suchgebietes erreicht wird. Dann wird die Suche mit verringerter Verschiebung fortgesetzt. Die Anzahl der Suchschritte ist vom Bildmaterial abhängig und variiert auch mit der Wahl der Abstandswerte. Für ein Suchgebiet mit  $D_{max} = 6$  ergeben sich maximal  $Q_{fs} = 21$  Fehlermaßberechnungen.

Das "three step" genannte Verfahren [7] liefert, im Gegensatz zum oben erwähnten 2D-logarithmic Verfahren, nach genau drei Suchschritten einen Vektor. Dazu wird eine achter Nachbarschaft in jedem Schritt untersucht und die Schrittweite in jedem dieser Schritte reduziert, Abb. 3.1.2. Es wird im allgemeinen in Schritten mit



Distanzwerten von 3, 2 bis 1 Pixel gesucht, wodurch eine Suchbereichsgröße von 6 Pixeln in jede Richtung erreicht wird. Bei diesem Verfahren werden zur Bestimmung eines Bewegungsvektors  $Q_{fs} = 3 \cdot 8 + 1 = 25$  Berechnungen des Fehlermaßes benötigt. Eine Vergrößerung des Suchbereiches bei gleicher Auflösung kann nur durch die Einführung weiterer Suchschritte erreicht werden.

Ein weiteres Verfahren "conjugate direction" [6], welches Abb. 3.1.3 zeigt, sucht (in der vereinfachten Form nach Musmann [21]) zuerst das Fehlerminimum in horizontaler, und danach, ausgehend von diesem Punkt, in vertikaler Richtung. Hierbei werden Verschiebungen in diagonalen Richtungen nicht genügend berücksichtigt. Die Zahl der Fehlermaßberechnungen ist vom gefundenen Bewegungsvektor abhängig. Zur Bestimmung des Bewegungsvektors werden maximal  $Q_{cd} = 2 \cdot (D_{max} + 1) + 1 = 15$  dieser Berechnungen benötigt.

### 3.2 Modifizierte volle Suche

#### 3.2.1 Fehlermodell

Alle oben aufgezeigten schnellen Suchverfahren gehen von einem monotonen Fehlermodell aus: Unter der Annahme, daß das Fehlerkriterium von der Position der wahren Verschiebung in alle Richtungen monoton ansteigt, konvergieren schnelle Suchverfahren auf das absolute Minimum. Anschaulich müßte das Fehlerkriterium dargestellt in der Ebene der möglichen Verschiebungen einen Trichter ergeben, dessen Minimum bei der Position des wahren Bewegungsvektors liegt. In Abb. 3.2.1 ist der Verlauf des Fehlermaßes  $MQF$  über der Verschiebung für einen realen Bildblock dargestellt. Es ist deutlich erkennbar, daß bei derartigem Fehlerverhalten keine Konvergenz der schnellen Suchverfahren auf das absolute Minimum erwartet werden kann. Durch die Feinstruktur der Objekte und Rauschen im Bildsignal entstehen lokale Minima des Fehlersignales, auf die die schnellen Suchalgorithmen konvergieren. Durch eine Tiefpaßfilterung der Ausgangsbilder kann dieser Effekt zwar verringert werden, die Genauigkeit der Bewegungsschätzung wird dadurch jedoch beeinträchtigt.

Das absolute Fehlerminimum wird unabhängig vom Verlauf des Fehlerkriteriums über der Verschiebung durch eine volle Suche gefunden. Ein Suchbereich der Größe  $D_{max} = 11$ , welcher bei zeitlicher Unterabtastung um den Faktor 3 mindestens erforderlich ist, benötigt jedoch  $Q_{fs} = 23 \cdot 23 = 529$  Suchpositionen. An jeder Position ist die Differenz der Bildpunkte des Blockes und eine Akkumulation der Differenzen im Falle des MAD-Kriteriums zu berechnen. Dies ergibt für eine Blockgröße von  $16 \cdot 16$  Punkten 256 Subtraktionen und Additionen. Insgesamt sind pro Bild ca. 107 Millionen Operationen nötig um das Vektorfeld zu bestimmen.

#### 3.2.2 Vektorquantisierung der Bewegungsvektoren

Durch Kombination der vollen Suche mit einer anschließenden Vektorquantisierung der Bewegungsvektoren kann der oben genannte Rechenaufwand erheblich reduziert werden: Entsprechend der Eigenschaft des menschlichen visuellen Systems, daß schnelle Bewegungen nur ungenau aufgelöst werden können, wird nur in einer kleinen Umgebung der Nullposition jede Position abgesucht, in größerem Abstand vom Nullpunkt wird ein groberes Raster verwendet. Dieser Zusammenhang ist in Abb. 3.2.2 für einen Suchbereich von  $\pm 6$  Punkten mit einem Bereich von  $\pm 2$  Punkten mit voller Auflösung dargestellt. Im Coder wurde ein Bereich von  $\pm 3$  Punkten mit voller Auflösung bei einem Gesamtsuchbereich von  $\pm 11$  Punkten verwendet. Die Auflösung wird im äußeren Bereich halbiert, so daß nur noch ein Viertel aller Positionen abzusuchen ist. Der Rechenaufwand reduziert sich entsprechend auf  $Q_{mfs} = 137$  Suchpositionen. Da für die jetzt entfallenden Vektorpositionen kein Codewort im Codebuch reserviert werden muß, reduziert sich auch die benötigte Datenrate für die Übertragung der Bewegungsvektoren. Das Verfahren wird auch durch die Verteilungsdichtefunktion der Vektorzustände, die in Abb. 3.2.3 dargestellt ist, gerechtfertigt. Größere Vektoren werden nur mit wesentlich geringerer Wahrscheinlichkeit zugewiesen als kleine Vektoren. Das hier geschilderte Verfahren der synthetischen Codebuchgenerierung kann durch eine iterative Optimierung des Codebuches noch weiter verbessert werden.

#### 3.2.3 Segmentierung

Die Anwendung eines Bewegungsschätzverfahrens auf unbewegte Bildbereiche mit wenig Struktur führt schon bei schwach verrauschten Bildsignalen zur Bestimmung von vom Nullvektor verschiedenen Bewegungsvektoren. Da die unbewegten Bildbereiche, vor allem im Bildhintergrund, bei den für Bildtelefonanwendung als typisch anzusehenden "Kopf-Schulterbildern" einen Anteil von 50% bis 80% oder mehr haben können, führt dieses Verhalten zu einem starken Anstieg der für die Übertragung der Bewegungsvektoren benötigten Datenrate. Bei der Bildrekonstruktion bewirken diese Bewegungsvektoren im Original nicht vorhandene Aktivität und Fluktuationen im Bildhintergrund, die überaus störend in Erscheinung treten. In Abb. 3.2.4 ist ein solches Vektorfeld dargestellt. Die Einführung einer Schwelle für den Start der Bewegungsvektorsuche führt zu einer weitgehenden Unterdrückung der durch Rauschen verursachten, vom Nullvektor verschiedenen Bewegungsvektoren in unbewegten Bildbereichen. Die Suche wird dabei nur dann gestartet, wenn das Fehlermaß an der Position des Nullvektors diese Schwelle übersteigt. Dies ist bei durch Rauschen hervorgerufenen Bildänderungen und richtiger Dimensionierung der Schwelle nicht der Fall. Die Schwelle sollte adaptiv an die Qualität des Bildmaterials angepaßt werden.

Da für die betroffenen Bildbereiche die Suche nach dem Bewegungsvektor entfällt, wird bei typischem Bildmaterial der Rechenaufwand für die Bewegungskompensation um den Faktor 2 bis 3 reduziert. Auch bei Bildblöcken, die diese Schwelle überschreiten, soll nur dann ein von Null verschiedener Bewegungsvektor bestimmt werden, wenn der Nutzen dieses Vektors größer ist als die "Kosten" für die Übertragung des Bewe-

gungsvektors. Der Nullvektor wird deshalb nur dann ersetzt, wenn das Fehlermaß, nach Ausführung der Bewegungskompensation, das des Nullvektors um diesen sogenannten Bonus unterschreitet. Abb. 3.2.5 zeigt das entsprechende Vektorfeld nach Einführung dieser beiden Modifikationen. Es ist deutlich erkennbar, daß die rauschbedingten Vektoren (vergleiche Abb. 3.2.4) im Bildhintergrund verschwinden. Die Güte der Prädiktion in

tatsächlich bewegten Bereichen wird durch diese Maßnahmen nicht beeinträchtigt.  
Die Bevorzugung des Nullvektors durch die Einführung des Bonus hat den weiteren Vorteil, daß die Form der Verteilungsdichtefunktion der Vektorzustände günstig beeinflußt wird. Die extreme Ungleichverteilung der Vektorzustände mit bis zu 50% Nullvektoranteil kann durch Anwendung einer Entropiecodierung (Huffmancode) vorteilhaft ausgenutzt werden.

### 3.2.4 Vergleich verschiedener Bewegungsschätzalgorithmen

Zur Beurteilung von Bewegungsschätzalgorithmen müssen die erzeugten Vektorfelder in zweierlei Hinsicht überprüft werden:

- Genauigkeit der Vektoren: Die Leistung des Prädiktionsfehlerbildes muß minimiert werden.
- Zuverlässigkeit der Vektoren: Die Bewegungsvektoren müssen die wahre Bewegung der Objekte widerspiegeln, um zur Interpolation bzw. Extrapolation ausgelassener Frames einsetzbar zu sein.

Während der zweite Punkt Gegenstand von Kapitel 6 ist, sollen hier die verschiedenen Algorithmen auf ihre Fähigkeit, das Differenzsignal nach Bewegungskompensation — also den Prädiktionsfehler — zu minimieren, untersucht werden. Als Kriterium wird die Leistung des bewegungskompensierten Differenzbildes in Relation zur Leistung des nicht bewegungskompensierten Differenzbildes gesetzt. In die Berechnung werden nur Blöcke einbezogen, die zum bewegten Vordergrund gehören, die oben beschriebene Segmentierung mit Hilfe der Schwellenoperationen wird also für alle Verfahren gleichermaßen verwendet.

In Abb. 3.2.6 ist dieses Verhältnis für verschiedene schnelle Suchverfahren und für die volle Suche über den untersuchten Bildern der Testsequenz "MISS AMERICA" aufgetragen. Ein y-Achsenwert von 0,10, zum Beispiel, bedeutet eine Reduktion der Leistung des Prädiktionsfehlerbildes um 90% durch Einsatz der Bewegungskompensation. Der Verlauf des Kriteriums für die volle Suche stellt das erreichbare Optimum dar und spiegelt die Bewegungskompensierbarkeit der Bilder wider: Ein großer Wert, selbst im Fall der vollen Suche, deutet auf Bildbereiche, die nicht aus dem vorhergehenden Bild stammen und daher nicht wegkompensiert werden können, z. B. Schließen der Augen. Alle gezeigten Verfahren haben einen Suchbereich von  $\pm 6$  Pixeln, die Blockgröße beträgt  $8 \times 8$  Bildpunkte. Die ermittelten Vektoren haben jeweils Pel-Genauigkeit. Die Leistungsfähigkeit der Verfahren korrespondiert zum jeweiligen Rechenaufwand: Während das 3-Step Verfahren und das "logarithmic search" Verfahren relativ ähnliche Ergebnisse liefern — mit leichten Vorteilen für das 3-Step-Verfahren —, verhält sich das "conjugate gradient"-Verfahren deutlich schlechter. Keines der Verfahren erreicht bemerkenswerterweise in auch nur einem Bild die Leistungsfähigkeit der vollen Suche, was die obige Aussage unterstreicht, daß das Fehlermodell der schnellen Verfahren der Praxis nicht hinreichend entspricht.

Abb. 3.2.7 vergleicht das volle Suchverfahren mit der oben beschriebenen modifizierten vollen Suche (volle Suche mit Vektorquantisierung) und dem 3-step Verfahren, als bestem Vertreter der schnellen Suchverfahren. Die Leistungsfähigkeit der modifizierten Suche approximiert die Leistung der vollen Suche in vielen Fällen besser als das 3-step Verfahren. Es existieren jedoch Situationen, in denen das künstlich generierte Codebuch der vollen Suche mit Vektorquantisierung die eigentlich nötigen Bewegungsvektoren für minimalen Fehler nicht enthält. Dies ist zum Beispiel bei den Bildnummern 75 oder 96 der Fall. Dann erzielt sogar das 3-step-Verfahren, welches je mit Pel-Genauigkeit arbeitet, bessere Ergebnisse. Eine Erweiterung der modifizierten vollen Suche, welche auch diesen Nachteil vermeidet, wird in Kap. 3.2.5 vorgestellt.

Schließlich wurde der Einfluß der Größe des Suchbereiches untersucht. Dazu wurde ein Suchbereich der Größe  $\pm 6$  Punkte mit einem Suchbereich der Größe  $\pm 11$  Punkte verglichen. In beiden Fällen wurde eine volle Suche durchgeführt. Wie Abb. 3.2.8 zeigt ist ein Suchbereich von  $\pm 6$  Punkten nicht ausreichend. Eine weitere Vergrößerung über  $\pm 11$  hinaus ergab für die Testsequenz "MISS AMERICA" keine weitere wesentliche Verbesserung.

### 3.2.5 Hierarchische modifizierte volle Suche

Mit dem oben beschriebenen Algorithmus der modifizierten Suche erhält man Vektoren, die nur in einem kleinen Bereich um den Nullpunkt Pel-Genauigkeit aufweisen. Im Bereich außerhalb beträgt die Auflösung sogar nur 2 Pel, so daß das im letzten Schritt stets mit 1 Pel Genauigkeit arbeitende 3-step-Verfahren in bestimmten Situationen bessere Ergebnisse liefert. Dies ist nämlich immer dann der Fall, wenn vom Bildmaterial Bewegungsvektoren gefordert werden, die sich nicht im Codebuch befinden. Untersuchungen [18] haben darüber hinaus gezeigt, daß eine Genauigkeit von 0,5 Pel anzustreben ist. Die mit noch feinerer Auflösung erreichbaren Verbesserungen sind gering und werden durch den nötigen Aufwand nicht gerechtfertigt.

Die durch die modifizierte volle Suche erzielten Bewegungsvektoren werden nun als erste Schätzung betrachtet und durch Nachschalten einer zweiten Bewegungsschätzung verbessert. Die zweite Stufe arbeitet mit einem sehr kleinen Suchbereich aber mit sub-Pel-Genauigkeit. Die Berechnungen der Bildpunkte zwischen den Rasterpunkten wird mit Hilfe einer bilinearen Interpolation durchgeführt. Die Kombination der beiden Suchalgorithmen kann als hierarchisches Verfahren bezeichnet werden. Abb. 3.2.9 zeigt das Prinzip des Verfahrens.

Wie Abb. 3.2.9 zeigt, werden zunächst die Verschiebungen von Teilbildbereichen in einem in Teilsuchgebiete gegliederten Suchgebiet ermittelt. Jedes Teilsuchgebiet wird durch eine Ersatzverschiebungsposition charakterisiert. In Abb. 3.2.9 ist die Ersatzverschiebungsposition durch den "Bewegungsvektor nach 1. Stufe" dargestellt,

d. h. alle möglichen Bewegungsvektoren innerhalb des Teilsuchgebiets werden durch die Ersatzverschiebungsposition beschrieben (Vektorquantisierung). Er führt zum Mittelpunkt des Teilsuchgebiets des 1. Suchschritts. Für den 2. Suchschritt wird das im 1. Schritt gefundene Teilsuchgebiet, welches die größte Ähnlichkeit bezüglich des Teilbildbereichs mit dem zeitlich davor liegenden Bild aufweist, für eine feinere Suche mit erhöhter örtlicher Auflösung (0,5 pel) ausgewählt. Die Suchgebiete im Ausführungsbeispiel nach Abb. 3.2.9 haben eine rechteckige Umrandung. Diese Umrandung kann natürlich in beliebiger Weise abgewandelt werden (Trapeze, Dreiecke, ...). Insbesondere kann eine Anpassung an das vorliegende Bildmaterial und an die menschliche Augenempfindlichkeit erfolgen.

Die mit dem hierarchischen Verfahren erzielten Ergebnisse sind in der Abb. 3.2.10 dargestellt. Es ist zu erkennen, daß durch Anwendung der hierarchischen Suche die Fehlerleistung gegenüber der vollen Suche mit 1 Pel Genauigkeit deutlich verringert werden konnte und die Leistungsfähigkeit einer vollen Suche mit 0,5 Pel Genauigkeit gut approximiert wird. Durch Anwendung der modifizierten vollen Suche als ersten Stufe ist der dazu nötige Rechenaufwand wesentlich geringer als bei anderen Verfahren mit 0,5 Pel Auflösung, da die Suche der zweiten Stufe auf einen sehr kleinen Suchbereich beschränkt ist. Das hierarchische Verfahren benötigt 186 Suchpositionen bei einem Suchbereich der Größe  $\pm 11$  Pel. Für die volle Suche mit 0,5 Pel Genauigkeit und gleichem Suchbereich sind 2025 Suchpositionen zu überprüfen. Durch Verwendung der hierarchischen vollen Suche kann also eine Einsparung an Rechenzeit um den Faktor 11 bei annähernd gleicher Leistungsfähigkeit erzielt werden. Die modifizierte volle Suche als erste Stufe bietet darüberhinaus den Vorteil, daß eine Annahme über das streng monotone Verhalten des Fehlerkriteriums bei Verschiebung nicht notwendig ist. Abb. 3.2.11 zeigt an Hand eines Bildbeispiels die Leistungsfähigkeit der hierarchischen Suche: Die Prädiktionsfehlerleistung bei Codierung des Bildes # 84 der Sequenz "MISS AMERICA" konnte deutlich reduziert werden. In Abb. 3.2.12 ist das Prädiktionsfehlerbild bei Codierung des gleichen Bildes aber Verwendung verschiedener Suchalgorithmen dargestellt. Zur Verdeutlichung der Unterschiede wurden die Prädiktionsfehlerbilder um den Faktor 2 verstärkt. Der subjektive Bildeindruck korrespondiert zum Fehlerkriterium in Abb. 3.2.10.

### 3.3 Codierung der Bewegungsvektoren

Unter dem Gesichtspunkt der Effizienz werden verschiedene Verfahren der Bewegungsvektorcodierung verglichen [22]. Untersucht wurden eine volle Suche bzw. modifizierte volle Suche mit Pel-Genauigkeit der Vektoren und einem Suchbereich von  $\pm 6$  Bildpunkten. Bei der modifizierten vollen Suche beträgt der Bereich maximaler Auflösung  $\pm 2$  Bildpunkte. Den Angaben der Gesamtbitzahl pro Bild für die Bewegungsvektoren liegt das CIF-Format und eine Blockgröße von  $16 \cdot 16$  zugrunde.

#### 3.3.1 Ein-/zwei-dimensionale Codierung der Bewegungsvektoren

Zunächst soll untersucht werden ob es günstiger ist, die Vektorkomponenten getrennt oder gemeinsam zu codieren. Für die beiden ausgewählten Verfahren wurde sowohl eine Entropiecodierung der Vektorkomponenten als auch eine zweidimensionale Entropiecodierung durchgeführt. Es zeigte sich daß die Summe der Entropien der Einzelkomponenten in jedem Fall größer ist als die Entropie des zweidimensionalen Vektors.

#### 3.3.2 Differenzcodierung der Bewegungsvektoren

Da die bewegten Objekte im Bild größer als die zur Bewegungsschätzung verwendeten Bildblöcke sind, kann zwischen den Bewegungsvektoren benachbarter Blöcke eine gewisse Korrelation erwartet werden. Die für drei verschiedene Sequenzen gemessenen Korrelationen zeigt Tabelle 3.3.1. Man erkennt, daß die speziell für Sequenzen mit wenig Bewegung ("ALEXIS") sehr gering sind. Es wurden drei verschiedene Codierv Verfahren verglichen und in Tabelle 3.3.2 gegenübergestellt:

- Zweidimensionale Entropiecodierung der Vektoren.
- Interframe Prädiktion aus zeitlich vorhergehendem Vektor an gleicher Position
- Interframe Prädiktion aus oberen und linken Nachbarvektoren

Während die zeitliche Prädiktion leichte Vorteile zeigt, weist die räumliche Prädiktion außer bei "TREVOR" nur Nachteile auf. Da durch die Verwendung von Vektordifferenzen die Anzahl der möglichen Vektorzustände stark erhöht wird, werden die Verbesserungen meist durch einen schlechteren Code kompensiert. Für den Fall sehr starker Bewegung und speziell bei Kamerabewegungen (Pan) kann das Verfahren der örtlichen Prädiktion der Bewegungsvektoren jedoch erhebliche Vorteile bringen. Diese Fälle sind aber als nicht-typisch für die hier betrachtete Anwendung anzusehen, daher sollen diese Codierv Verfahren nicht weiter verfolgt werden.

#### 3.3.3 Quadtree-Codierung des Bewegungsvektorfeldes

Das mit dem in Abschnitt 3.2 vorgestellten Verfahren erzeugte Vektorfeld zeigt, daß großen zusammenhängenden Bildbereichen im Bildhintergrund Nullvektoren zugeordnet werden. Ebenso sind Bereiche mit von Null verschiedenen Vektoren meist zusammenhängend. Diese Tatsache soll durch Verwendung eines quaternären Baumes ausgenutzt werden. Abb. 3.3.3 zeigt das Prinzip. Enthält das Bild von Null verschiedene Bewegungsvektoren wird eine Unterteilung des Bildes in vier Unterbilder vorgenommen. Diese Untergliederung wird sukzessive fortgesetzt, bis ein Unterbild genau durch einen Bewegungsvektor repräsentiert wird. Auf diese Art und Weise erreicht man die Zusammenfassung möglichst großer Bereiche mit ausschließlich Nullvektoren und deren

gemeinsamer Codierung durch ein Bit.

In der Tabelle 3.3.4 ist dieses Quadtree-Verfahren der Bewegungsvektorcodierung an Hand der Sequenz "MISS AMERICA" dem Verfahren der zweidimensionalen Entropiecodierung der Vektoren gegenübergestellt.

Durch das Quadtree-Codiervorgehen kann eine effizientere Codierung der Bewegungsvektoren erreicht werden. Obwohl für den Baum zusätzliche Information übertragen werden muß, entfällt die Übertragung einer großen Zahl von Nullvektoren (20% — 65%). Das Verfahren hat aber zusätzlich den Vorteil, daß bei abklingender Bewegung eine zunehmende Datenrate für die Übertragung von Restfehlern zur Verfügung steht. Im Extremfall der Standbildübertragung genügt ein Bit um dem Empfänger mitzuteilen, daß alle Vektoren Nullvektoren sind. Die gesamte Datenrate kann also zur Verbesserung des Bildinhaltes (z. B. Erhöhung der Ortauflösung) herangezogen werden. Dies führt zu einem progressiven Update zu Beginn bewegungsarmer Sequenzphasen.

#### 4. Codierung des Prädiktionsfehlerbildes

Für die Codierung des Prädiktionsfehlerbildes sind bis jetzt fast ausnahmslos Verfahren der intraframe Codierung von Grauwertbildern verwendet worden. Diese gehen meist von einer Transformation, zum Beispiel DCT, aus. Dabei wird die hohe örtliche Korrelation natürlicher Grauwertbilder ausgenutzt. Durch Transformation werden die Daten in ein neues Koordinatensystem abgebildet, in dem die hier vorliegenden Koeffizienten der Transformation nur noch schwache Korrelationen aufweisen. Anders ausgedrückt: Die in einem Bildblock relativ gleichmäßig verteilte Leistung wird auf wenige Spektralkoeffizienten konzentriert. Für eine Approximation des Bildblockes genügt daher die Übertragung einiger weniger leistungstragender Koeffizienten. Die in der Literatur veröffentlichten Verfahren unterscheiden sich durch die Auswahl und die Quantisierung der zu codierenden Spektralkoeffizienten.

Eine vollständige Dekorrelation kann durch eine Karhunen-Loève Transformation (KLT) erzielt werden. Allerdings ist dazu die Berechnung der Kovarianzmatrix des Bildprozesses erforderlich, woraus über eine Eigenwerttransformation die Transformationsmatrix gewonnen wird. Unter der für natürliche Bildvorlagen meist mit hinreichender Genauigkeit zutreffenden Annahme, daß der bilderzeugende Zufallsprozeß durch einen Markov-Prozeß erster Ordnung modelliert werden kann, kann gezeigt werden, daß die KLT approximiert [23]. Dies gilt jedoch nur solange, wie der Korrelationskoeffizient in der für natürliche Bildvorlagen typischen Größenordnung von 0,95 — 0,98 liegt. In [24] ist gezeigt worden, daß die DCT für schwach korrelierte Daten weniger gute Eigenschaften aufweist, welches die Eignung der DCT zur Codierung der Prädiktionsfehlerbilder in Frage stellt. Aus den vorangegangenen Überlegungen heraus soll nun zunächst das Prädiktionsfehlersignal statistisch analysiert [25] und mit natürlichen Bildvorlagen verglichen werden. Anschließend wird der durch Anwendung der DCT erreichte Codiergewinn untersucht. Durch Einführung einer Strukturcodierung kann eine bessere Datenkompression bei gleichzeitig verringertem Rechenaufwand erzielt werden. Der Algorithmus wird schließlich im Rahmen eines neuen Coderkonzeptes für eine Rate von 64 kbit/s vorgestellt.

##### 4.1 Statistische Eigenschaften des Prädiktionsfehlerbildes

Verfahren zur Datenkompression von Bildern nutzen die starke Korrelation benachbarter Bildpunkte aus. Abb. 4.1.1 zeigt die Autokorrelationsfunktion, ermittelt aus 150 Bildern die Sequenz "MISS AMERIKA", für jeweils horizontale und vertikale Richtung. Eine kombinierte Darstellung zeigt Abb. 4.1.2. Aus den Abbildungen geht hervor, daß zwei benachbarte Punkte einen Korrelationskoeffizienten von 0,98 haben.

Im Vergleich zum Originalbild zeigen die Abb. 4.1.3 und 4.1.4 die entsprechenden Autokorrelationsfunktionen für die Prädiktionsfehlerbilder, ebenfalls ermittelt aus der Sequenz "MISS AMERIKA" unter Verwendung des oben vorgestellten Bewegungskompensationsalgorithmus. Deutlich ist die nun wesentlich geringere Korrelation der Bildpunkte zu erkennen. Der Korrelationskoeffizient benachbarter Punkte beträgt 0,49.

##### 4.1.1 Autokorrelationsfunktion eines Hochpaßsignals

Das Prädiktionsfehlersignal soll nun als Hochpaßkomponente modelliert werden. Eine Schaltung zur Erzeugung dieser Hochpaßkomponente zeigt Abb. 4.1.5. Für die Autokorrelationsfunktion der Differenz zweier Signale gilt:

$$e(k,l) = s(k,l) - g(k,l) \quad (4.1.1)$$

$$\varphi_{ee}(k,l) = \varphi_{ss}(k,l) + \varphi_{gg}(k,l) - \varphi_{sg}(k,l) - \varphi_{gs}(k,l) \quad (4.1.2)$$

$\varphi_{ss}(k,l)$ : Autokorrelation des Originalprozesses, gemessen oder Modellierung durch Markov-Prozeß erster Ordnung

$\varphi_{gg}(k,l) = \varphi_{ss}(k,l) \cdot \varphi_{hh}^E(k,l)$ ; Wiener-Lee-Beziehung mit der Impulsautokorrelationsfunktion des Tiefpaßfilters:  $\varphi_{hh}^E(k,l)$  (4.1.3)

$$\varphi_{sg}(k,l) = \varphi_{ss}(k,l) \cdot h(k,l) \quad (4.1.4)$$

$$\varphi_{gs}(k,l) = \varphi_{ss}(k,l) \cdot h(-k, -l) \quad (4.1.5)$$

Unter der Voraussetzung symmetrischer Tiefpässe erhält man identische Kreuzkorrelationsterme:

$$h(k,l) = h(-k, -l) = \varphi_{ss}(k,l) = \varphi_{gs}(k,l) \quad (4.1.6)$$

Gleichung (4.1.2) vereinfacht sich entsprechend zu:

$$\varphi_{ee}(k,l) = \varphi_{ss}(k,l) + [\varphi_{ss}(k,l) \cdot \varphi_{hh}^E(k,l)] - 2 \cdot [\varphi_{ss}(k,l) \cdot h(k,l)] \quad (4.1.7)$$

Wählt man für  $h(k,l)$  einen idealen Tiefpaß

$$H(u,v) = \text{rect}(u/U) \cdot \text{rect}(v/V) \circ \circ h(k,l) = UV \text{sinc}(nUk) \cdot \text{sinc}(nVl) \quad (4.1.8)$$

erhält man:

$$\varphi_{ee}(k,l) = \varphi_{ss}(k,l) - [\varphi_{ss}(k,l) \cdot (UV \text{sinc}(nUk) \cdot \text{sinc}(nVl))] \quad (4.1.9)$$

#### 4.1.2. Autokorrelationsfunktion eines Markov-Prozesses erster Ordnung

Zur Berechnung von  $\varphi_{ss}(k,l)$  modelliert man das Zufallssignal  $s(m,n)$  als autoregressiven Prozeß. Ein AR-Prozeß oder auch Markov-Prozeß erster Ordnung ist durch einen "weißen" und zeitdiskreten Zufallsprozeß  $z(n)$  und die folgende Beziehung gegeben (zur Vereinfachung erfolgt die Ableitung hier 1-dimensional, zur Erweiterung auf mehr Dimensionen muß zwischen separierbaren und isotropen Modellen unterschieden werden, siehe [36]):

$$x(n) = z(n) + \rho \cdot x(n-1) \text{ mit } \rho = \text{Autokorrelationskoeffizient} \quad (4.1.10)$$

$$= z(n) + \sum_{j=1}^{\infty} \rho^j \cdot z(n-j)$$

Für eine Verschiebung um  $k$  gilt:

$$x(n+k) = z(n+k) + \sum_{j=1}^{\infty} \rho^j \cdot z(n+k-j) \quad (4.1.11)$$

Damit errechnet sich die Autokorrelationsfunktion  $\varphi_{xx}(k)$  eines Markov-Prozesses erster Ordnung:

$$\begin{aligned} \varphi_{xx}(k) &= E[x(n) \cdot x(n+k)] \\ &= \varphi_{zz}(k) + \sum_{j=1}^{\infty} \rho^j \cdot \varphi_{zz}(k+j) + \sum_{j=-n}^{\infty} \rho^j \cdot \varphi_{zz}(k-j) + \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \rho^i \rho^j \cdot \varphi_{zz}(k+i-j) \end{aligned} \quad (4.1.12)$$

Mit der Autokorrelationsfunktion des "weißen" Prozesses

$$\varphi_{zz}(k) = \sigma_z^2 \cdot \delta(k) \quad (4.1.13)$$

erhält man:

$$\varphi_{xx}(k) = \sigma_x^2 \cdot \delta^{|k|} = \sigma_x^2 \cdot \exp(|k| \cdot \ln(\sigma)) \quad (4.1.14)$$

Die Varianz des Markov-Prozesses folgt aus der Varianz des Ausgangsprozesses unter Berücksichtigung des Korrelationskoeffizienten zu:

$$\sigma_x^2 = \varphi_{xx}(0) = \frac{\sigma_z^2}{1-\rho^2} \quad (4.1.15)$$

Durch Normierung auf die Varianz erhält man die normierte Autokorrelationsfunktion:

$$\rho_{xx}(k) = \exp(|k| \cdot \ln(\rho)) \quad (4.1.16)$$

#### 4.1.3 Autokorrelationsfunktion eines hochpaßgefilterten Markov-Prozesses

Die Korrelation benachbarter Bildpunkte entsteht durch die zeitliche Korrelation des zeitkontinuierlichen Videosignals. Die Berechnung der Autokorrelationsfunktion eines tief- bzw. hochpaßgefilterten Markovprozesses erster Ordnung soll zur Vereinfachung für zeitkontinuierliche Signale durchgeführt werden. Die Autokorrelationsfunktion eines zeitkontinuierlichen Markov-Prozesses erster Ordnung ist gegeben durch:

$$\rho_{xx}(\tau) = \exp(-|\tau|/T) \quad (4.1.17)$$

Es ergibt sich folgende Beziehung zwischen der örtlichen Korrelation  $\rho$  und der Zeitkonstante  $T$ :

—X—X—X—X—→ scan-Zeile

5  $\langle k = 1 \rangle$

$$k \cdot \ln(\rho) = -1/T \cdot \Delta t \text{ mit } \Delta t = \text{Zeit zwischen Abtastwerten} \quad (4.1.18)$$

10 
$$T = \frac{-\Delta t}{k \cdot \ln(\rho)} \text{ mit } \Delta t \approx 150 \text{ ns und } \rho = 0,98$$

$$T = 7,42 \mu\text{s}$$

15 Aus der Autokorrelationsfunktion  $p_{xx}(\tau)$  erhält man das Leistungsdichtespektrum

20 
$$\Phi_{xx}(f) = \int_{-\infty}^{\infty} e^{-1/T} e^{-j2\pi f\tau} d\tau \quad (4.1.19)$$

25 
$$= \frac{2T}{1 + (2\pi fT)^2}$$

Das Leistungsdichtespektrum wird nun ideal hochpaßgefiltert mit der Grenzfrequenz  $f_0$ :

30 
$$\Phi_{xx}(f) = \frac{2T}{1 + (2\pi fT)^2} \text{ für } f \geq f_0 \quad (4.1.20)$$

$$\Phi_{xx}(f) = 0 \quad \text{für } 0 < f < f_0$$

35 Die Autokorrelationsfunktion des hochpaßgefilterten Markovprozesses ergibt sich durch inverse Fouriertransformation. Die resultierende Autokorrelationsfunktion ist die Faltung der Autokorrelationsfunktion des Markov-Prozesses mit  $\text{si}(nf_0\tau)$ . Unter der Bedingung  $f_0 \gg 1/T$ , welche hier erfüllt ist, ergibt sich die folgende Näherung für die Autokorrelationsfunktion des bandbegrenzten Prozesses (Ableitung der Näherung siehe Anhang):

40 
$$p_{xx}^{HP}(\tau) \approx \cos(2nf_0\tau) + 2nf_0\tau \cdot (S(2nf_0\tau) - n/2) \quad (4.1.21)$$

In Abb. 4.1.6 ist der gemessene Verlauf der Autokorrelationsfunktion des Prädiktionsfehlerbildes dargestellt. Zum Vergleich wurde das Originalbild mit einem FIR-Filter der Größe  $7 \times 7$  tiefpaßgefiltert und vom gleichen Bild subtrahiert. Der Verlauf der Kurven verdeutlicht die Gültigkeit der Modellannahme. Die Autokorrelationsfunktion des ideal hochpaßgefilterten Markov-Prozesses erster Ordnung nach Gl. 4.1.21 ist in Abb. 4.1.7 für  $2 \cdot nf_0\tau = k$  dargestellt. Mit  $k = \text{Pixelabstand}$  und dem oben für  $\tau$  angenommenen Wert  $\tau = 150 \text{ ns}$  erhält man eine untere Grenzfrequenz von  $f_0 \approx 1 \text{ MHz}$ . Der Unterschied zum Verlauf der gemessenen Autokorrelationsfunktion in Abb. 4.1.6 erklärt sich durch Restkorrelation im Prädiktionsfehlerbild durch unzureichende Bewegungskompensation. Dies könnte durch Gewichtsfaktoren bei der Modellierung des Prädiktionsfehlerbildes nach Abb. 4.1.5 berücksichtigt werden: Originalbild und tiefpaßgefiltertes Originalbild werden vor der Differenzbildung multiplikativ gewichtet, um im Differenzbild einen bestimmten Originalbildanteil zu erhalten.

Da der Gewinn der in der Bildcodierung verwendeten Prädiktions- und Transformationsverfahren auf einer hohen Korrelation der Bildpunkte beruht, soll im folgenden gezeigt werden, daß diese Verfahren zur Codierung des Prädiktionsfehlersignals wenig geeignet sind.

## 4.2 Anwendung der Transformationscodierung auf schwach korrelierte Daten

Die schwache Korrelation der Bildpunkte des Prädiktionsfehlerbildes läßt erwarten, daß durch Transformationscodierung der Daten keine nennenswerte Leistungskonzentration erzielt werden kann [24]. Zur Abschätzung des Codierungsgewinns werden im folgenden verschiedene Kriterien zur Beurteilung der Effizienz von Transformationen untersucht.

### 4.2.1 Eigenwerte der Kovarianzmatrix

65 Ereignisse eines Zufallsprozesses sind genau dann unkorreliert, wenn die Kovarianzmatrix des Zufallsprozesses Diagonalform hat, also alle Kovarianzen verschwinden. Für den Sonderfall eines Gaußprozesses sind die Ereignisse sogar statistisch unabhängig. Um zu einer diagonalförmigen Kovarianzmatrix zu gelangen, kann man

eine sogenannte Eigenwert-Transformation durchführen, in der Literatur wird diese Transformation "Karhunen-Loève" oder auch "Hotelling-Transformation" genannt. Dabei bilden die Eigenvektoren die Spalten der Transformationsmatrix. Diese KLT stellt demnach das erreichbare Optimum für eine Transformationscodierung dar. Man kann zeigen, daß die DCT für die Klasse der natürlichen Bildvorlagen, welche durch einen autoregressiven Prozess erster Ordnung modelliert werden können, die KLT mit größer werdender Blockgröße asymptotisch approximiert. Da für die KLT kein schneller Algorithmus existiert und die Transformationsmatrix relativ rechenaufwendig an die jeweilige Statistik angepaßt werden muß, spricht man bei der DCT auch von einer schnellen KLT. Durch Sortierung der Eigenvektoren in der Transformationsmatrix, entsprechend dem Betrag der zugehörigen Eigenwerte, erreicht man bei der KLT eine Leistungskonzentration auf die Spektralkoeffizienten mit kleinstem Index. Der Gewinn durch Leistungskonzentration ist dabei umso größer je größer der Unterschied im Betrag der Eigenwerte ist. In Abb. 4.2.1 wird der Verlauf der Eigenwerte für die Statistik des Prädiktionsfehlerbildes und im Vergleich dazu für die Originalsequenz gezeigt. Der nur durch Transformationsverfahren kein wesentlicher Gewinn zu erzielen ist.

Für den Fall der DCT wird in Abb. 4.2.2 der Verlauf der Varianzen der Spektralkoeffizienten verglichen. Für das Prädiktionsfehlersignal erhält man im Vergleich zu natürlichem Bildmaterial eine relativ geringe Abnahme der Varianzen mit steigender Koeffizientenordnung. Die akkumulierten spektralen Varianzen in Abhängigkeit von der Anzahl der übertragenen Spektralkoeffizienten zeigt Abb. 4.2.3. Für eine natürliche Bildvorlage wird die Varianz des Bildblockes schon durch Berücksichtigung weniger Koeffizienten fast vollständig repräsentiert. Im Falle des Prädiktionsfehlersignals sind für das Erreichen der gleichen Wechselleistung wesentlich mehr Koeffizienten erforderlich.

#### 4.2.2 Entropie

Ein weiteres Effizienzmaß für die Transformation kann mit der Entropie gewonnen werden. Dazu wird die Entropie erster Ordnung sowohl im Bildbereich, als auch im Spektralbereich berechnet. Im Spektralbereich wird für den Fall, daß alle Spektralkoeffizienten gleiche Werte annehmen die Entropie maximal. Dies kann als vollständiges Fehlen der Leistungskonzentrationseigenschaften angesehen werden. Andererseits wird die Entropie im Spektralbereich für die Konzentration der Leistung auf einen Koeffizienten minimal. Im Bildbereich erhält man für die Entropie erster Ordnung

$$H(x) = - \sum_{k=1}^L p_k \log_2 p_k, \quad (4.2.1)$$

mit  $p_k$  als der Wahrscheinlichkeit, daß die Zufallsvariable  $x$  den Wert  $r_k$  annimmt. Der Vorrat an Zeichen ergibt sich aus der Quantisierung der Bildpunkte mit 8 bit in diesem Fall zu  $L = 2^8$ . Der Wert von  $H(x)$  ist ein Maß für den mittleren Informationsgehalt eines Bildpunktes des Prädiktionsfehlerbildes. An Hand von 50 Prädiktionsfehlerbildern aus der Testsequenz "MISS AMERIKA" wurde der Wert für die Entropie bestimmt:  $H(x) = 3,82$  bit/pixel.

Zum Vergleich wird nun eine blockweise diskrete Cosinus Transformation mit einer Blockgröße von  $8 \times 8$  Bildpunkten auf das Prädiktionsfehlerbild angewendet. Aus den Wahrscheinlichkeiten für das Auftreten der Amplituden der Spektralkoeffizienten ermittelt man nun die Entropie im Spektralbereich  $H(u)$ . Geht man von einer optischen und für jeden Spektralkoeffizienten getrennten Codierung aus erhält man die Entropie durch Mittelung der Einzelentropien für jeden Koeffizienten:

$$H'(u) = - \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{m=1}^M p_k^{(i,j)} \log_2 p_k^{(i,j)} \quad (4.2.2)$$

Der Spektralkoeffizient  $u^{(i,j)}$  nimmt mit der Wahrscheinlichkeit  $p_k^{(i,j)}$  einen Wert aus dem möglichen Wertebereich vom  $[(1-M) \leq u \leq M]$  an. Dabei muß  $M$  hinreichend groß gewählt werden. Der mögliche Wertebereich der Spektralkoeffizienten ist zum Beispiel in [14] angegeben.

Für die oben erwähnte Sequenz wurden gemessen:

$$h(u) = 3,59 \text{ bit/pixel};$$

$$H'(u) = 3,45 \text{ bit/pixel}.$$

Diese Ergebnisse widersprechen der allgemeinen Annahme, daß durch Transformationscodierung des Prädiktionsfehlerbildes eine Datenkompression erreicht werden kann. Vielmehr wird die Entropie durch Transformation oft vergrößert. Auch die getrennte Codierung jedes einzelnen Spektralkoeffizienten erreicht nur eine geringe Verbesserung.

#### 4.2.3 Momente höherer Ordnung

Von Strobach wurden in [32] ein Verfahren vorgeschlagen, die geraden Momente höherer Ordnung als Indikator für die Leistungskonzentration zu verwenden. Die Motivation für ein derartiges Effizienz-Kriterium ergibt sich aus der Leistungsinvarianz der verwendeten Transformationen. Die Momente zweiter Ordnung sind also im Original- Spektralbereich identisch. Für die Transformationsmatrix  $T$  erhält man für den Datenvektor  $x$

den Vektor der Spektralkoeffizienten  $y$  (Zur Vereinfachung der Darstellung wird nur der eindimensionale Fall abgeleitet, die Erweiterung auf zwei Dimensionen folgt direkt):

$$y = Tx \quad (4.2.3)$$

$$y^T y = x^T T^T x = x^T x - \text{Leistungsinvarianz} \quad (4.2.4)$$

oder in Momentendarstellung

$$\sum_{l=1}^N y^2(l) = \sum_{l=1}^N x^2(l) \quad (4.2.5)$$

Dies ist für die Momente höherer Ordnung nicht mehr der Fall:

$$\sum_{l=1}^N y^k(l) \neq \sum_{l=1}^N x^k(l) \text{ für } k > 2 \text{ und } k \text{ gerade} \quad (4.2.6)$$

Man kann erwarten, daß bei vorhandener Leistungskonzentration die Momente höherer Ordnung im Spektralbereich wesentlich größere Werte annehmen als die entsprechenden Momente im Originalbereich. Aus dem Verhältnis der Momente höherer Ordnung im Original und im Spektralbereich erhält man also ein Maß für den Grad der erreichten Leistungskonzentration. Dieser Leistungskonzentrationskoeffizient  $\rho_k$  in Abhängigkeit von der Ordnung der Momente  $k$  ist folgendermaßen definiert:

$$\rho_k = \rho_k(x, T) = \frac{\sum_{l=1}^N y^k(l)}{\sum_{l=1}^N x^k(l)} \text{ für } k > 2 \text{ und } k \text{ gerade} \quad (4.2.7)$$

Abb. 4.2.4 zeigt die Verteilung dieses Kriteriums für das Moment vierter Ordnung und die Testsequenz "MISS AMERIKA". Für den Bereich  $\rho_k > 1$  kann eine Transformation vorteilhaft eingesetzt werden. Die Abbildung zeigt jedoch eine Häufung um den Wert  $\rho_4 = 2$ . Für natürliche Bildvorlagen wird typischerweise ein Wert von  $\rho_4 \approx 63$  erreicht. Daraus kann man ersehen, daß die Transformation in vielen Fällen eine Expansion der Daten zufolge hat und selbst im Falle der Kompression nur mäßige Leistungskonzentration erreicht wird.

#### 4.2.4 Rate-Distortion-Funktion

Eine Abschätzung des erreichbaren Gewinnes durch Transformationscodierung von Hochpaßbildern wurde von Amor [31] mit Hilfe der Rate-Distortion-Funktion durchgeführt. Da, wie oben gezeigt, das vorliegende Bildmaterial durch ein Hochpaßbildsignal modelliert werden kann, sind die Ergebnisse hier gültig. Danach kann mit einer Transformationscodierung eine Einsparung von 0,3 bit/Bildpunkt gegenüber PCM-Codierung erzielt werden. Dieses Ergebnis steht in Übereinstimmung mit den obigen Effizienzuntersuchungen.

Auf der Grundlage der oben genannten Ergebnisse wird im folgenden eine Strukturcodierung eingeführt, die eine effiziente Codierung des Prädiktionsfehlerbildes ermöglicht.

#### 4.3 Strukturcodierung

Zur Veranschaulichung ist in Abb. 4.3.1 ein typisches Prädiktionsfehlerbild abgebildet. Als Approximation der Verteilungsdichtefunktion ist in Abb. 4.3.2 das aus 50 Prädiktionsfehlerbildern gemessene Amplitudenhistogramm dargestellt. Zur Verwendung der Codierung in einen Bewegtbildcoder mit niedriger Datenrate, wird eine Codierung des Prädiktionsfehlerbildes mit  $\approx 0,8$  bit/Bildpunkt angestrebt. Dieser Wert soll bei subjektiv guter Qualität erreicht werden. Dies ist deshalb besonders wichtig, da die codierten Bildbereiche zur Prädiktion des darauffolgenden Frames herangezogen werden. Eine Codierung mit geringer Qualität würde verhindern, daß das im Empfänger rekonstruierte Bild bei geringer werdender Bewegung das Originalbild sukzessive approximiert. Darüberhinaus kann die Wahrnehmbarkeit von Rekonstruktionsfehlern durch Bewegungseffekte erhöht sein.

Die Helligkeitswerte der Bildpunkte werden unabhängig voneinander und linear in 256 Stufen quantisiert. Diese Auflösung ist nötig, damit im Bereich höchster Kontrastempfindlichkeit des menschlichen Sehapparates keine Quantisierungsfehler wahrnehmbar werden. Diese Kontrastempfindlichkeit hängt aber sehr stark von der Ortsfrequenz der beobachteten Struktur ab. Dieser Zusammenhang wurde z. B. in [34] untersucht und ist in Abb. 4.3.3 dargestellt. Diese sogenannte Modulationsübertragungsfunktion zeigt ein deutliches Maximum der Empfindlichkeit für Ortsfrequenzen um 1 bis 5 Zyklen pro Grad. Der Verlauf der Modulationsübertragungsfunktion ist stark von der Grundhelligkeit abhängig. In allen Fällen ist jedoch ein starker Rückgang der Empfindlich-



keit für höhere Ortsfrequenzen zu beobachten. Daher ist eine hohe Amplitudenaufösung mit 8 bit/Bildpunkt im Falle des Prädiktionsfehlersignals, welches ja, wie oben gezeigt, als Hochpaßkomponente mit hinreichender Genauigkeit modelliert werden kann, nicht erforderlich.

Die lineare Quantisierung des Originalbildes ist wegen der annähernden Gleichverteilung der Amplituden gerechtfertigt. Diese Voraussetzung trifft jedoch im Falle des vorliegenden Bildmaterials nicht zu, wie Abb. 4.3.2 zeigt. Die Form der Verteilungsdichtefunktion kann bei linearer Quantisierung durch Entropiecodierung der Ersatzwerte ausgenutzt werden. Dadurch wird die Verteilungsredundanz entfernt; allerdings liegen die Einsparungen weit oberhalb der zu erreichenden Zieldatenrate.

Für das Qualitätskriterium "mittlerer quadratischer Fehler" ist von Max [28] ein Algorithmus zur Quantisierung mit minimalem Fehler angegeben worden. Die Berechnung der Ersatzwerte und Stufen erfolgt jedoch iterativ und ist mit beträchtlichem Aufwand verbunden. Die von Amor [31] durchgeführten Untersuchungen an Hochpaßkomponenten natürlicher Bildvorlagen haben zum Ergebnis, daß eine dreiwertige Quantisierung dann ausreichend ist, wenn die Quantisierererkennlinien adaptiv in mindestens vier Klassen umgeschaltet werden. Die oben gezeigte mögliche Modellierung des Prädiktionsfehlerbildes durch ein Hochpaßbild und die daraufhin durchgeführten Untersuchungen mit dreiwertiger Quantisierung bestätigen die Übertragbarkeit der Ergebnisse. Die Verwendung eines Nullersatzwertes bietet sich wegen der spitzen Form der Verteilungsdichtefunktion an. Die Bestimmung der positiven und negativen Ersatzwerte für jede der vier Quantisierererkennlinien erfolgt nach folgendem Algorithmus (siehe Abb. 4.3.4):

- \* Bestimmung von 3 Klassengrenzen an Hand der Fehlervarianz; die Unterteilung wird so vorgenommen, daß in jede Klasse ungefähr die gleiche Anzahl von Blockrealisationen fällt.
- \* Approximation der Verteilungsdichtefunktionen  $p(x)$  für jede Klasse durch Berechnung des Amplitudenhistogramms getrennt nach Klassen.
- \* Berechnung des Quantisierer-Ersatzwertes, der den mittleren quadratischen Quantisierungsfehler minimiert. Es wird eine symmetrische Form der Verteilungsdichtefunktion vorausgesetzt. Die Leistung des Fehlers  $N_q$  berechnet sich aus der Amplitude des Fehlersignals  $x$  und bei bekannter Verteilungsdichtefunktion  $p(x)$  für die Schwelle  $u$  und den Ersatzwert  $v$  zu:

$$N_q = 1/n \cdot \left[ \sum_{x=0}^{u-1} p(x) \cdot x^2 + \sum_{x=u}^{x_{max}} p(x) \cdot (x - v)^2 \right] \quad (4.3.1)$$

Die Bedingung für minimalen Quantisierungsfehler lautet:

$$\frac{\partial N_q}{\partial v} = 0 \quad (4.3.2)$$

Daraus ergibt sich der optimale Ersatzwert  $V_{opt}$  zu:

$$V_{opt} = \frac{\sum_{x=u}^{255} p(x) \cdot x}{\sum_{x=u}^{255} p(x)} \quad (4.3.3)$$

Für die Berechnung wird zunächst eine Schwelle  $u = 1$  angesetzt.

- \* Bestimmung der optimalen Schwelle  $u_{opt}$  für den gewählten Ersatzwert  $v_{opt}$ , so daß der Quantisierungsfehler minimal wird. Dazu wird ausgehend von der Schwelle  $u = 1$  diese solange vergrößert, bis die Leistung des resultierenden Fehlersignals ihr Minimum erreicht.

Die Berechnung der Quantisiererersatzwerte und Schwellen wird für jedes Bild erneut vorgenommen, so daß eine optimale Adaption an die Bildstatistik erzielt wird. Für die Verwendung der Strukturcodierung in einem Bewegtbildcoder ist diese Tatsache besonders wichtig: Bei großen Bildveränderungen kann der Bildaufbau durch Verwendung großer Ersatzwerte rasch erfolgen. Bei abklingender Bewegung werden kleinere Ersatzwerte berechnet, was die asymptotische Annäherung an die Originalbildqualität garantiert. Die den vier Klassen entsprechenden Verteilungsdichtefunktionen sind am Beispiel eines Bildes in Abb. 4.3.5 dargestellt. In der Tabelle sind die Daten zusammengestellt:

	Varianzen des Fehlerbildes	Klasse	Grenzen	Ersatzwerte
5	$0 \leq \rho^2 < 764$ $764 \leq \rho^2 < 996$ $996 \leq \rho^2 < 1737$ $1737 \leq \rho^2 < 3155$ $3155 \leq \rho^2 < \text{dct-Schwelle}$ $\text{dct-Schwelle} \leq \rho^2$	— 1 2 3 4 #	— -3,0,3 -3,0,3 -4,0,4 -7,0,7 DCT codierter Originalbildblock	0,0,0 -5,0,5 -5,0,5 -7,0,7 -12,0,12
10				

Die Adaption in vier Klassen erfordert es allerdings, diese Overhead-Informationen dem Empfänger mitzuteilen. Dies benötigt ca. 40 bit/Bild, kann also vernachlässigt werden. Alternativ könnten die Verteilungsdichtefunktionen durch verallgemeinerte Gaußfunktionen modelliert werden. Nach der Anpassung der Formfaktoren kann eine einmalige, optimale Bestimmung der Schwellen und Ersatzwerte erfolgen. Dadurch ist zwar keine nennenswerte Einsparung an Datenrate zu erzielen, die Struktur des Senders vereinfacht sich jedoch erheblich, was für eine Hardwarerealisierung wichtig ist. Die Auswahl der Quantisierungskennlinie erfolgt jeweils für einen  $8 \times 8$  Punkte großen Bildblock und muß ebenfalls zusätzlich übertragen werden.

Zur Übertragung der dreiwertigen Bilder soll nun untersucht werden, ob sich bestimmte, besonders häufig vorkommende Muster herausbilden. Ein derartiges Verfahren wurde bereits von Shibata [35] zur Codierung von Videosignalen vorgeschlagen. Durch eine orthogonale Transformation wurde eine Analyse der vorherrschenden Muster durchgeführt. Es zeigte sich, daß bei natürlichem Bildmaterial und Verwendung der Hadamard-, bzw. Slant-Transformation nur ungefähr 1000 verschiedene Muster bei einem  $16 \times 16$  Bildpunkte großen Block auftraten. Die Vorgehensweise kann als ein Sonderfall der Codebuchgenerierung für eine Vektorquantisierung angesehen werden: Aus einem feststehenden Codebuch, gegeben durch die Basisbilder der verwendeten Transformation, wird ein Untercodebuch der am häufigsten auftretenden Muster selektiert.

Wegen der dreiwertigen Quantisierung und bei Untergliederung in hinreichend kleine Unterblöcke der Größe  $N \times N$  Bildpunkte, kann wegen der begrenzten Anzahl der vorkommenden Möglichkeiten zunächst ein vollständiges Codebuch erstellt werden. Im Gegensatz zu Verfahren der Transformationscodierung wo der Transformationsgewinn mit steigender Blockgröße zunimmt, bieten große Blöcke wegen der mangelnden Korrelation der Bildpunkte hier keine Vorteile. Die Verwendung kleinerer Blöcke bietet im Gegenteil den Vorteil, daß Blockstrukturen im Bild wegen der reduzierten Empfindlichkeit des menschlichen Sehapparates für höhere Ortsfrequenzen nicht oder nicht so leicht wahrgenommen werden. Ordnet man jedem Blocktyp ein Codewort zu, benötigt man  $3^{N \times N}$  Codeworte. Schon für  $N = 3$  beträgt diese Zahl 19 683, so daß für  $N$  nur der Wert 2 in Frage kommt. Die Auftrittshäufigkeit der nun möglichen 81 verschiedenen Blocktypen ist in Abb. 4.3.6 dargestellt. Der Index der Typen berechnet sich wie folgt:

Positioncode:

1	3
9	27

Faktor: negativer Ersatzwert: \*1  
Null-Ersatzwert: \*0  
positiver Ersatzwert: \*2

Beispiel:

+	0
0	-

ergibt Index = 29

Selbst unter Ausnutzung der Verteilung der Auftrittshäufigkeit durch Entropiecodierung ergibt sich eine Rate von 1,4 bit/Bildpunkt. Durch Eingrenzung der Typenvielfalt soll nun die erforderliche Rate vermindert werden, ohne große subjektive Qualitätseinbußen in Kauf nehmen zu müssen. Die im Abb. 4.3.6 mit '^' markierten Blocktypen enthalten zugleich negative und positive Ersatzwerte. Ihre Auftrittshäufigkeit ist relativ gering. Überführt man diese Typen in Typen, welche nur positive und nur negative Ersatzwerte enthalten, kann die Zahl der möglichen Typen ohne wesentliche Qualitätseinbuße auf 31 gesenkt werden. Die Überführung in andere Blocktypen geschieht nach den folgenden Regeln:

- \* Blöcke, die gleichviele positive und negative Ersatzwerte haben, werden durch Blöcke mit Index = 0, d. h. nur Nullersatzwerte, repräsentiert.
- \* Blöcke, welche mehr positive als negative Ersatzwerte haben, werden durch positive und Nullersatzwerte dargestellt.
- \* Wie oben, aber für eine Überzahl negativer Ersatzwerte.

Durch diese Maßnahmen kann die Zieldatenrate von 0,8 bit/Bildpunkt erreicht werden. Abb. 4.3.7 zeigt die Auftrittshäufigkeit der verbleibenden 31 Blocktypen. Unter Verwendung eines Quadrates innerhalb eines Update-Blockes der Größe  $8 \times 8$  kann die Dominanz der Blocktypen mit ausschließlich Nullersatzwerten ausgenutzt

und eine weitere Reduzierung der Datenrate erreicht werden.

#### 4.4 Coder-Konzept für 64 kbit/s

Die oben dargelegten Untersuchungen lassen die Strukturcodierung für eine Verwendung in einem 64 kbit/s Codekonzept geeignet erscheinen. Im folgenden sollen zunächst die Besonderheiten herausgestellt werden, die das hier gewählte Gesamtkonzept neben der Verwendung einer neuen Codierung gegenüber anderen Ansätzen auszeichnen:

- \* Es wird nicht das gesamte Fehlerbild zum Empfänger übertragen. Entsprechend der Wirksamkeit der Bewegungskompensation und der zur Verfügung stehenden Restdatenrate werden nur die Blöcke zum Empfänger übertragen (Update), welche das größte Fehlermaß aufweisen. Zur Signalisierung, Overhead-Codierung, wird ein Quadtree verwendet.
- \* Es wird keine einheitliche Codierung des Prädiktionsfehlersignals vorgenommen. An Hand des blockweisen Fehlermaßes wird die Wirksamkeit der Bewegungskompensation klassifiziert. Dabei ist es in Bereichen, wo die Bewegungskompensation versagt, zum Beispiel durch neue Bildteile, die in die Szene kommen, effektiver, einen DCT codierten Originalbildblock zum Empfänger zu übermitteln, als den Bereich durch Updates zu codieren.
- \* Das Coderkonzept verwendet keinen Puffer zur Erzielung einer konstanten Datenrate. Vielmehr wird für jeden Frame exakt die zur Verfügung stehende Rate ausgenutzt. Der Vorteil dieser Vorgehensweise ist neben der einfachen Struktur ohne Rückkopplung und Buffersteuerung vor allem die konstante Codierqualität. Die konstante Codierqualität der als "Update" übertragenen Bildteile ist eine wichtige Voraussetzung für die Wirksamkeit der Prädiktion des nachfolgenden Bildes.

Ausgehend von einem üblichen hybriden Coder/Decoder Konzept (Abb. 4.4.1.a) ist das Coderkonzept gemäß der Erfindung in einem detaillierten Blockschaltbild in Abbildung 4.4.1.b dargestellt. Einzelne Blöcke sollen im folgenden eingehender vorgestellt werden:

##### 4.4.1 Bewegungsschätzung, Prädiktion

Mit Hilfe der Bewegungsvektoren für jeden Bildblock wird ein Schätzbild erstellt. Die Bewegungsvektoren werden nur aus der Luminanzinformation ermittelt, aber auch zur Kompensation der beiden Chrominanzkomponenten eingesetzt. Dies ist möglich, da Chrominanzänderungen ohne entsprechende Veränderungen in der Luminanz äußerst selten sind. Durch Subtraktion des Originalbildes erhält man das Differenzbild. Für jeden Bildblock wird das Fehlermaß, mittlerer quadratischer Fehler (MQF), berechnet. Die Bewegungsvektorinformation wird vektorquantisiert. Die starke Dominanz von Null-Vektoren (mehr als 50% typischerweise) verschlechtert die Möglichkeit einer Entropiecodierung der Bewegungsvektoren, da für den Nullvektor minimal 1 Bit übertragen werden muß. Deshalb wird eine Quadtree-Codierung der Bewegungsvektoren vorgenommen. Nach Abzug der für die Übertragung der Bewegungsvektoren erforderlichen Datenrate erhält man die Restdatenrate, die zur Codierung des Prädiktionsfehlers zur Verfügung steht. Die Anzahl der mit dieser Restdatenrate übertragbaren Updateblöcke wird aus dem Codierungsergebnis des vorhergehenden Bildes geschätzt.

##### 4.4.2 Klassifikation der zu codierenden Fehlerblöcke

Mit Hilfe des MQF werden die Bildblöcke klassifiziert [33]:

- Typ A: In Bildbereichen in denen keine Bewegung stattgefunden hat, oder die Bewegung in Form starrer Translation erfolgte, welche durch die Schätzung praktisch vollständig kompensiert werden konnte, erhält man ein sehr kleines Fehlermaß. Blöcke vom Typ A brauchen daher nicht weiter betrachtet zu werden.
- Typ B: Hier erfolgte die Bewegung nicht streng nach dem Modell starrer Translation. So konnte das Differenzsignal durch die Bewegungsschätzung weitgehend reduziert werden, es verbleiben jedoch Restfehler, welche ein Update des Blockes erforderlich machen. Diese Restfehler äußern sich im Fehlerbild vor allem durch linienhafte Konturen infolge Verformung der Objekte. Eine geeignete Strukturcodierung für diese Daten wurde oben vorgestellt.
- Typ C: Dadurch, daß sich Bildteile stark veränderten, oder neue Bildteile in die Szene kamen, konnte die Bewegungskompensation nicht erfolgreich arbeiten. Beispiele für derartige Bildbereiche sind freiwerdender Hintergrund, Öffnen/Schließen oder Augen/des Mundes, Hände/Arme die durch Gestik den Bildrand passieren. In all diesen Fällen besteht der Prädiktionsfehler praktisch aus dem neuen Bildbereich, weist also die statistischen Eigenschaften eines natürlichen Bildes, wie zum Beispiel hohe örtliche Korrelation auf. Für diese Blöcke ist die Strukturcodierung weniger, die DCT jedoch sehr geeignet. Die Blöcke lassen sich einfach durch eine Schwellenoperation der Fehlervarianz detektieren. Man erhält bei Videotelefonsequenzen typischerweise 0—8 Blöcke vom Typ C pro Bild. Nach Abzug der dafür benötigten Datenmenge verbleibt die Restdatenrate für die Codierung der Blöcke vom Typ B. Für die Chrominanzinformation wird nur in Blöcke vom Typ A oder Typ B unterschieden, also keine DCT auf Chrominanzblöcke angewendet.

##### 4.4.3 Auswahl der mit Strukturcodierung zu übertragenden Blöcke

Aus der für das vorherige Bild benötigten Datenmenge pro Update-Block wird die maximal mögliche Anzahl

von Blöcken geschätzt, die ohne Überschreitung der pro Bild erlaubten Datenrate durch Strukturcodierung "upgedated" werden kann. Die Reihenfolge der Kandidaten-Blöcke wird durch den Betrag des Fehlermaßes bestimmt. Mit anderen Worten, die Blöcke, welche den größten Fehler aufweisen, werden bis zur Erreichung der maximalen Datenrate codiert übertragen. Damit ist sichergestellt, daß die Bildqualität jeweils das mögliche Optimum darstellt. Bei geringerem Bewegungsanteil finden sofort auch Blöcke mit kleinerem Fehlermaß Beachtung, so daß die Qualität wieder ansteigt. Dieser Automatismus befindet sich in Übereinstimmung mit dem visuellen System des Menschen. Man unterscheidet [26] beim menschlichen visuellen System ein sogenanntes "transient system" und ein "sustained system". Das "transiente system" ermöglicht die Erfassung schneller Bewegungen auf Kosten der Orts- und Amplitudenauflösung. Das "sustained system" kann auf Bildteile fokussieren und höchste Detailinformation liefern. Zu beachten ist allerdings, daß bei langsamen Bewegungen, denen das Auge durch Ausgleichsbewegungen folgen kann [27], das residente System dominiert.

Ein weiterer Vorteil der Coderstruktur liegt in der Verwendbarkeit des gleichen Algorithmus zur Übertragung von hochaufgelösten Standbildern. Sobald die Bewegungsvektoren Null werden, steht durch die Quadtree Codierung die gesamte Datenrate für Updates zur Verfügung. Dadurch wird schnell die höchstmögliche Auflösung erreicht.

Entsprechend dem oben vorgestellten Algorithmus und der Anzahl der möglichen Update-Blöcke wird eine Klasseneinteilung innerhalb der für die Übertragung ausgewählten Blöcke vom Typ B vorgenommen. Die Klassengrenzen werden dabei so gewählt, daß ungefähr die gleiche Anzahl von Blöcken in jeder der vier Klassen fallen. Die Bestimmung der Ersatzwerte erfolgt nach den Regeln der Quantisierung mit minimalem Fehler [28] durch Berechnung der Verteilungsdichtefunktion und der optimalen Ersatzwerte für jedes Bild neu. Dadurch wird ein progressiver Bildaufbau mit zunächst großen und dann immer kleiner werdenden Ersatzwerten erzielt.

#### 4.4.5 Quadtree-Codierung der Overheadinformation

Die Signalisierung, welche Blöcke übertragen werden erfolgt mit Hilfe eines Quadtree, der neben der Information keine Übertragung/Übertragung durch DCT/Übertragung durch Strukturcodierung auch die Klasseninformation für den letzten Fall beinhaltet. Für die Overhead-Information wird zur Ausnutzung der Verteilung ein Huffman-Code verwendet. Es ergeben sich folgende Codeworte:

Ereignis	Codewort	Darstellung in Abb. 4.4.2
kein Update	0	' '
Strukturcodierung Klasse 1	100	'0'
Strukturcodierung Klasse 2	101	'1'
Strukturcodierung Klasse 3	110	'2'
Strukturcodierung Klasse 4	1110	'3'
Codierung mit DCT	1111	

Ein zweiter Quadtree wird für die Chrominanz gebildet. Abb. 4.4.2 zeigt die Quadtree-codierte Signalisierungsinformation. Typischerweise werden für Anzeige und Klassifizierung der Blöcke, also die sogenannte Ortsinformation, ungefähr 300 Bits pro Frame für die Luminanz und 250 Bits für beide Chrominanzkomponenten benötigt.

#### 4.4.6 Strukturcodierung

Schließlich wird die eigentliche Strukturcodierung vorgenommen. In Abb. 4.4.3 ist das Ergebnis der Strukturcodierung gezeigt. Die ausgewählten Blöcke entsprechen der Signalisierungsinformation in Abb. 4.4.2. Die Rekonstruktion eines Frames der Sequenz "MISS AMERIKA" und "ALEXIS" codiert mit 64 kbit/s ist in Abb. 4.4.4 dargestellt.

#### Anhang

Das Leistungsdichtespektrum einer ideal hochpaßgefilterten Realisation eines Markov-Prozesses erster Ordnung beträgt:

$$\Phi_{xx}(f) = \frac{2T}{1 + (2\pi T f)^2} \text{ für } f \geq f_0 \quad (\text{A.1})$$

$\Phi_{xx}(f) = 0$  für  $0 < f < f_0$  Durch Integration über den Durchlaßbereich des Hochpaßfilters erhält man die

Autokorrelationsfunktion:

$$\varphi_{ss}(t) = 2 \cdot T \cdot \int_0^{\infty} \frac{1}{1 + (2\pi T f)^2} \cdot \cos(2\pi f t) df \quad (A.2)$$

$$= 2/T \int_0^{\infty} \frac{1}{1/T^2 + (2\pi f)^2} \cdot \cos(2\pi f t) df \quad (A.3)$$

Für den Fall, daß  $1/T^2 \ll (2\pi f_0)^2$  kann das Integral näherungsweise gelöst werden. Mit der Abschätzung 4.1.18 ergibt sich:  $T \approx 7,42 \mu s$ . Für  $f_0$  kann bei Bildtelefonsequenzen von einem Wert von  $\approx 1$  MHz ausgegangen werden, so daß die Ungleichung mit  $18,2 \cdot 10^9 \ll 39,5 \cdot 10^{12}$  in der Tat erfüllt ist.

$$\varphi_{ss}(t) \approx 2 \cdot T \cdot \int_0^{\infty} \frac{1}{(2\pi f)^2} \cdot \cos(2\pi f t) df \quad (A.4)$$

$$\approx \frac{t}{\pi T} \left( \frac{\cos(2\pi f_0 t)}{2\pi f_0 t} + \text{Si}(2\pi f_0 t) - \frac{\pi}{2} \right) \quad (A.5)$$

Der obige Ausdruck ergibt nicht  $\varphi_{ss}(0) = 1$ . Diese Normierung wird durch Division mit  $\varphi_{ss}(0)$  erreicht:

$$\rho_{ss}(\tau) \approx \cos(2\pi f_0 \tau) + (2\pi f_0 \tau) \cdot (\text{Si}(2\pi f_0 \tau) - \pi/2) \quad (A.6)$$

Die Funktion  $\rho_{ss}(\tau)$  ist in Abb. 4.1.7 dargestellt.

Literatur:

- [1] Widergren, R.D.: "Full Color Motion Video at 56 kbps: Military and Civil Applications", WIDCOM Inc, Globecom 85 Tagungsband, 1985, Vol.2, S. 681.
- [2] Netravali, A.N./Robbins, J.D.: "Motion Compensated Television Coding — Part I", The Bell System Technical Journal, Vol. 58, S. 631—670, März 1979.
- [3] Robbins, J.D./Netravali, A.N.: "Recursive Motion Compensation: A Review", in: Image Sequence Processing and Dynamic Scene Analysis, T.S. Huang, Ed., Berlin, Germany: Springer-Verlag, 1983, S. 76—103.
- [4] Stuller, J.A./Netravali, A.N.: "Transform Domain Motion Estimation", The Bell System Technical Journal, Vol. 58, No. 7, September 1979, S. 1673—1702.
- [5] Jain, J.R./Jain, A.K.: "Displacement Measure and its Applications in Interframe Image Coding", IEEE Trans. on Commun., vol. COM — 29, no. 12, Dezember 1981, S. 1799—1808.
- [6] Srinivasan, R./Rao, K.R.: "Motion Compensates Coder for Videoconferencing", IEEE Trans. on Commun., vol. COM — 35, no. 3, March 1987, S. 297—304.
- [7] Koga, T. et. al.: "Motion-compensated Interframe Coding for Video Conferencing", in NTC 81, Proc. pp. G5.3.1—G5.3.5, New Orleans, LA, Dec. 1981.
- [8] Kummerfeldt, G./May, F./Wolf, W.: "Coding Television Signals at 320 and 64 kbit/s", SPIE Conf, B594 Image Coding, Cannes, France, December 1985.
- [9] Ishiguro, T./Iinuma, K.: "Television Bandwidth Compression Transmission by Motion — Compensated Interframe Coding", IEEE Communications Magazine, November, 1982, S. 24—30.
- [10] Brusewitz, H.: "Variable Length Coding of Pictures in the Transform Domain", Report no. TRITA — TT — 8606, März 1987, Stockholm, Schweden.
- [11] Girod, B.: "The Efficiency of Motion — Compensating Prediction for Hybrid Coding of Video Sequences", IEEE Journal on Selected Areas in Communications, vol. SAC-5, no. 7, August 1987, S. 1140—1154.
- [12] Yamaguchi, H./Wada, M.: "64 kbps integrated Visual Telecommunication System", KDD Research & Development Laboratories, Japan.
- [13] Kamangar, F.A./Rao, K.R.: "Interfield hybrid coding of Component Color Television Signals. IEEE Trans. COM, 32—3, 1984.
- [14] Chen, W.H./Pratt, W.K.: "Scene Adaptive Coder" IEEE Trans. COM, 32-3, 1984, S. 225—232.
- [15] Miyahara, M./Kotani, K.: "Block Distortion in Orthogonal Transform Coding — Analysis, Minimization and Distortion Measure", IEEE Trans. COM, 1985, 33-1, S. 90—96.
- [16] Huang, T.S.: "Image Sequence Analysis", Springer Series in Information Science, Springer Verlag Berlin, Heidelberg, New York 1981.
- [17] Kuglin, C.D./Hines, D.C.: "The phase correlation image alignment method" Proc. of the IEEE, Conf. on Cybernetics and Society, Sept. 1975.
- [18] Girod, B./Joubert, F.: "Motion-compensates prediction with fractional pel accuracy for 64 kbit/s coding of moving video", Tagungsband Int. Workshop on 64 kbit/s coding of moving video, Hannover, Juni 1988.
- [19] Cafforio, L./Rocca, F.: "Methods for measuring small displacements of television images", IEEE Trans. on Information Theory, Vol. 12, Sept. 1976, S. 573—579.
- [20] Bergmann, H.C.: "Analysis of different displacement estimation algorithms for digital television signals", in

Image Sequence Analysis, edited by T.S. Huang, Berlin, Springer-Verlag, 1983, S. 215—234.

[21] Musmann, H.G./Pirsch, P./Grallert, H.-J.: "Advances in Picture Coding", Proc. of the IEEE, Vol. 73 No. 4. April 1985, S. 523—548.

[22] Schneider, L.: "Erstellung einer Programmbibliothek von Signalprozessorprogrammen und Implementierung in einem Multiprozessorsystem zur Stand- und Bewegtbildverarbeitung", Diplomarbeit, Inst. f. el. Nachrichtentechnik, RWTH Aachen, 1988.

[23] Jain, A.K.: "A fast Karhunen-Loeve transform for a class of stochastic processes", IEEE Trans. COM, 1976, 24-9, S. 1023—1029.

[24] Jain, A.K.: "A sinusoidal family of unitary transforms", IEEE Trans. PAMT, 1979, 1—4, S. 356—365.

[25] Gilge, M.: "A low bit-rate image sequence coder using signal adapted coding of the prediction error signal", Tagungsband Int. Workshop on 64 kbit/s coding of moving video, Hannover, Juni 1988.

[26] Kronander, T.: "Coding of image sequences with consideration to the human observer", Universität Linköping, Schweden, 1986, Thesis # 89.

[27] Girod, B.: "Ein Modell der menschlichen visuellen Wahrnehmung zur Irrelevanzreduktion von Fernsehlu-minanzsignalen", Reihe 10, Nr. 84, VDI-Verlag, 1988.

[28] Max, P.: "Quantizing for minimum distortion", IEEE Trans. IRE, IT-6, 1960, S. 7—12.

# Patentansprüche

1. Verfahren zum Reduzieren der Datenrate von digitalen Bilddaten mit folgenden Verfahrensschritten:

- a) Ermitteln von Bewegungsvektoren für Teilbildbereiche in zeitlich aufeinanderfolgenden Bildern einer Bildsequenz, derart, daß bezüglich zweier Bilder jeweils die Abweichung von der Position des unverschobenen Teilbildbereichs zur tatsächlichen Position des Teilbildbereiches in einem zeitlich folgenden Bild ermittelt und daraus der jeweilige Bewegungsvektor gebildet wird,
- b) Suchen der Verschiebungen der Teilbildbereiche in einem in Teilsuchgebiete gegliederten Suchgebiet derart, daß jedes dieser Teilsuchgebiete durch eine Ersatzverschiebungsposition charakterisiert wird,
- c) Auswählen jenes Teilsuchgebietes und damit der Ersatzverschiebungsposition, bei der die Ähnlichkeit der Teilbildbereiche zwischen zwei zeitlich aufeinanderfolgenden Bildern am größten ist,
- d) Aufbereiten der Ersatzverschiebungsposition zu einem Übertragungssignal oder zu einem Startwert für eine weitere Suche,

wobei der Schritt a) den Oberbegriff und die Schritte b) bis d) das Kennzeichen bilden.

2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, daß die Suche nach Ersatzverschiebungspositionen in dem zuvor ausgewählten Teilsuchbereich zur Bestimmung genauerer Ersatzverschiebungspositionen fortgesetzt wird, jedoch nun mit erhöhter örtlicher Auflösung als im ersten Suchschritt.

3. Verfahren nach einem der Ansprüche 1 oder 2, dadurch gekennzeichnet, daß aus den für jeden Teilbildbereich ermittelten Bewegungsvektoren ein Schätzbild erstellt wird, daß aus dem erstellten Schätzbild und dem Originalbild durch Differenzbildung ein Prädiktionsfehlerbild gebildet wird, daß eine Auswahl der zu übertragenden Fehlermenge des Prädiktionsbildes getroffen wird in Abhängigkeit eines Fehlerkriteriums, z. B. der Leistung des Prädiktionsfehlers, und daß die Signalisierung der ausgewählten Bildteile über einen "Quadtree" codiert wird.

4. Verfahren nach Anspruch 3, dadurch gekennzeichnet, daß das Prädiktionsfehlerbild klassifiziert wird entsprechend der Wirksamkeit einer Bewegungskompensation, daß in Abhängigkeit der Klassifizierung eine adaptive Grobquantisierung des Prädiktionsfehlers vorgenommen wird, wobei die Quantisierungsparemeter in Abhängigkeit der Klassifizierung gesteuert werden.

5. Verfahren nach Anspruch 4, dadurch gekennzeichnet, daß zur Grobquantisierung des Prädiktionsfehlers Strukturblöcke gebildet werden, die aus Gruppen von quantisierten Bildpunkten einer Nachbarschaft bestehen.

6. Verfahren nach Anspruch 4 oder 5, dadurch gekennzeichnet, daß für Bildbereiche, in denen eine Bewegungskompensation versagt, Originalbildinformation in dekorrelierter Form übertragen wird.

Hierzu 28 Seite(n) Zeichnungen

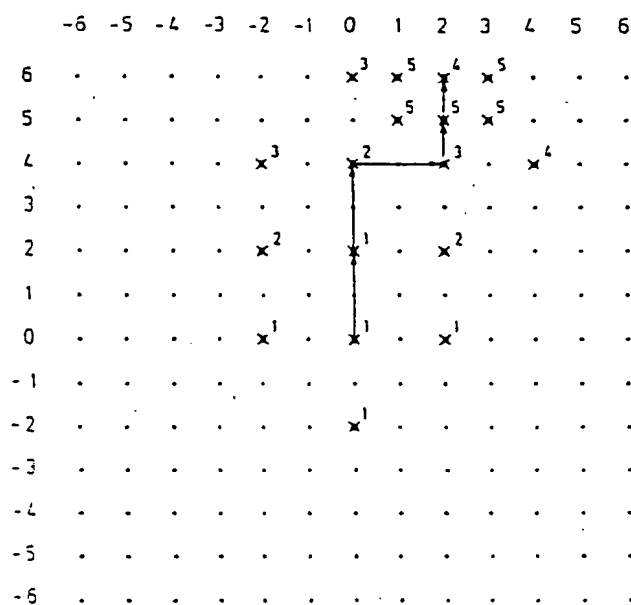


Fig. 1  
Abbildung 3.1.1:

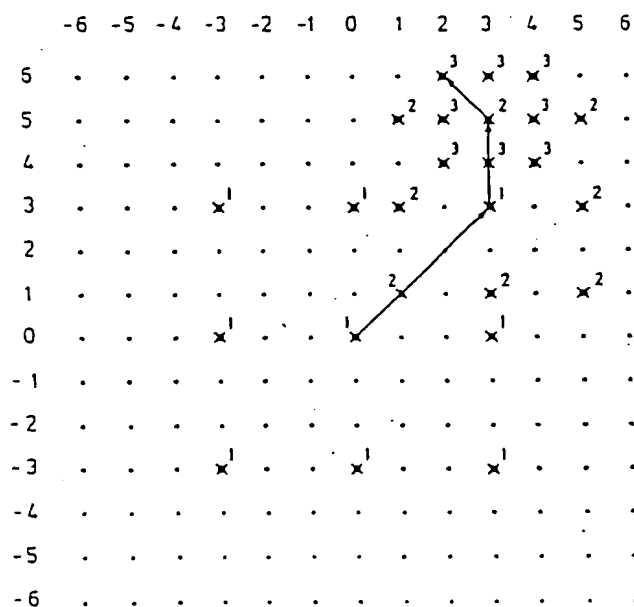


Fig. 2  
Abbildung 3.1.2:

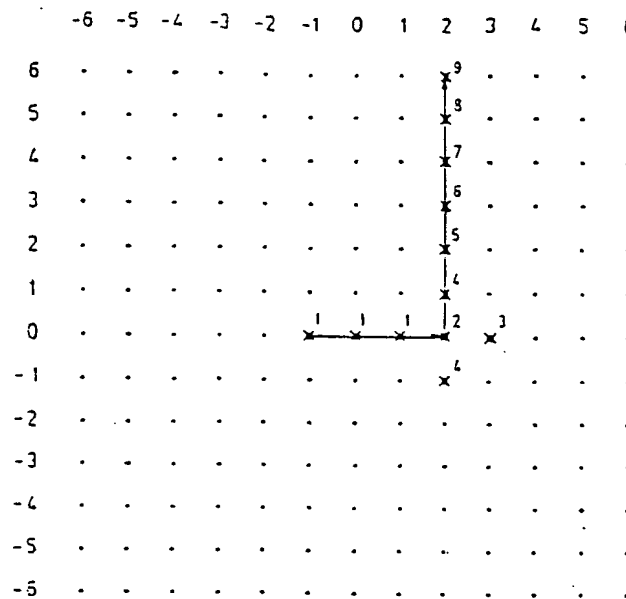


Fig 3  
Abbildung 3.1.3

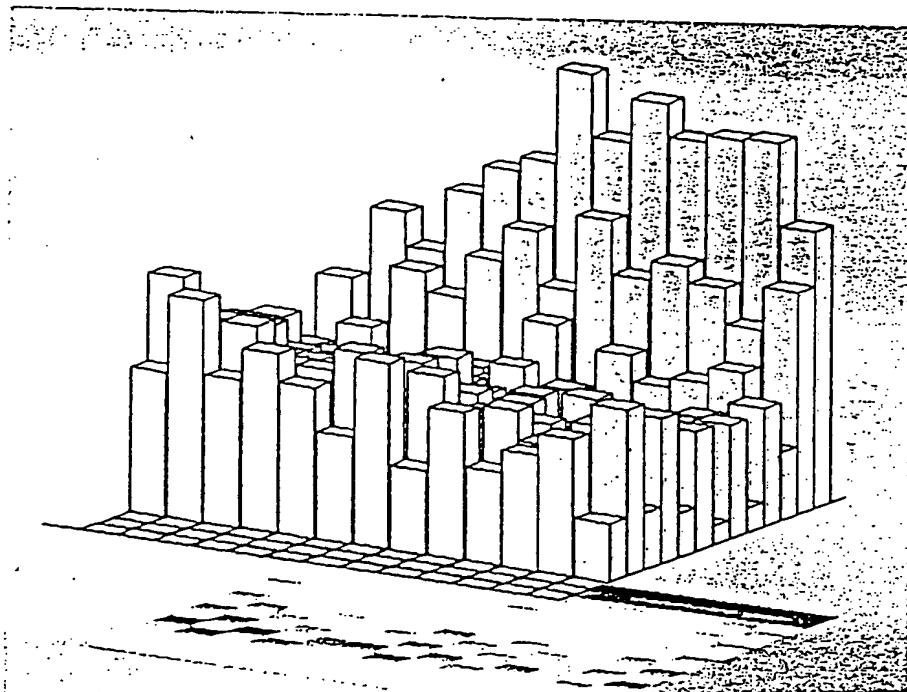


Fig. 4  
Abbildung 3.2.1



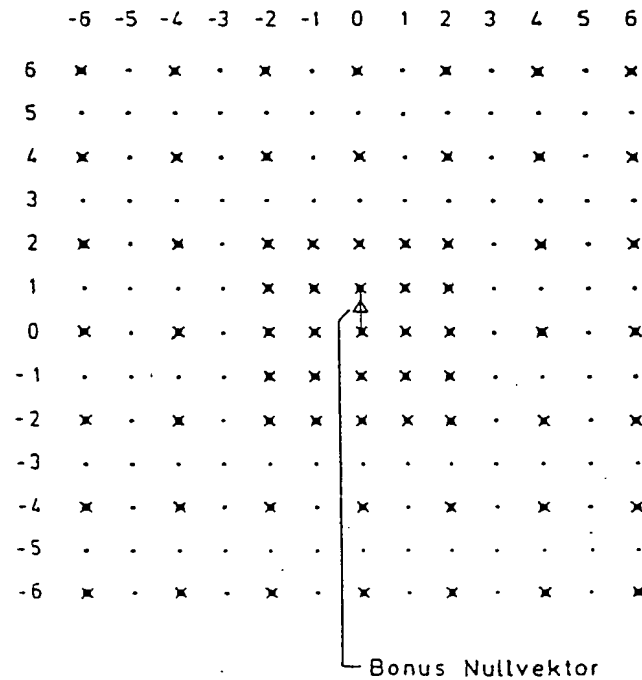


Fig. 5  
Abbildung 3.2.2

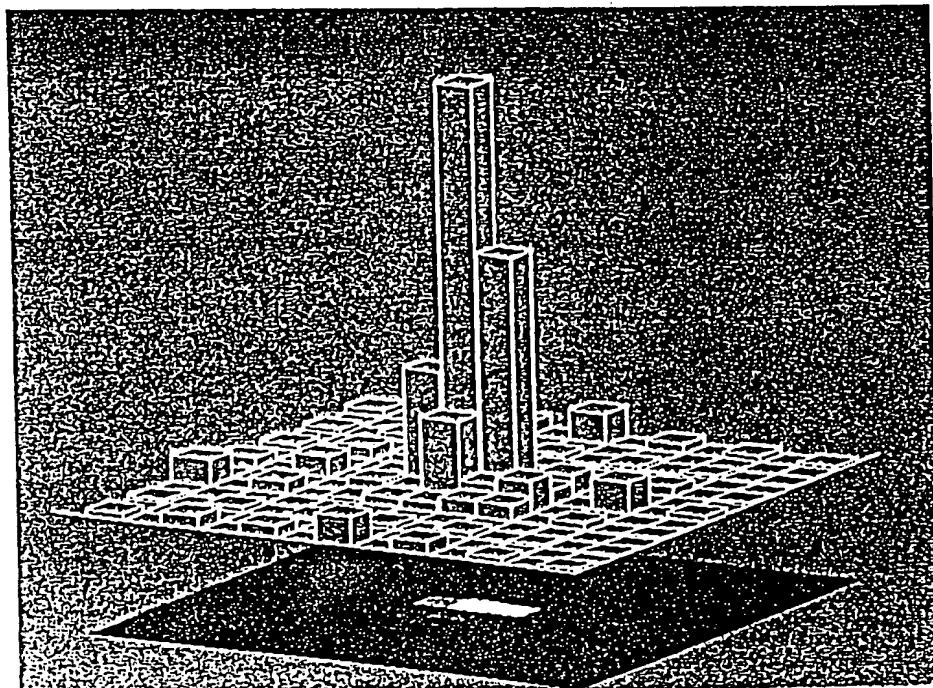


Fig. 6  
Abbildung 3.2.3

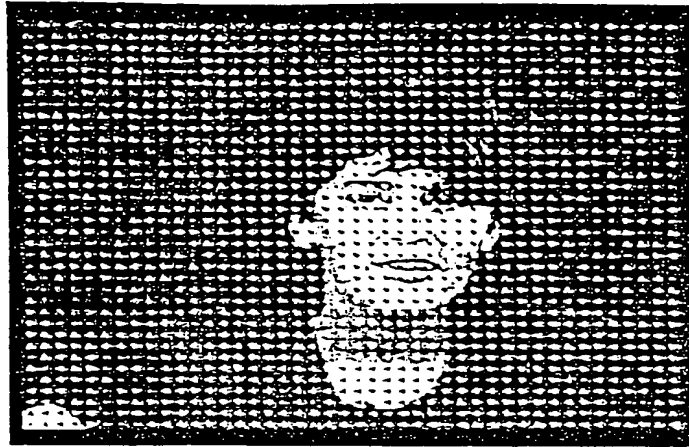


Fig. 7  
Abbildung 3.2.4



Fig. 8  
Abbildung 3.2.5

Leistung des Fehlerbildes mit/ohne Bewegungskompensation

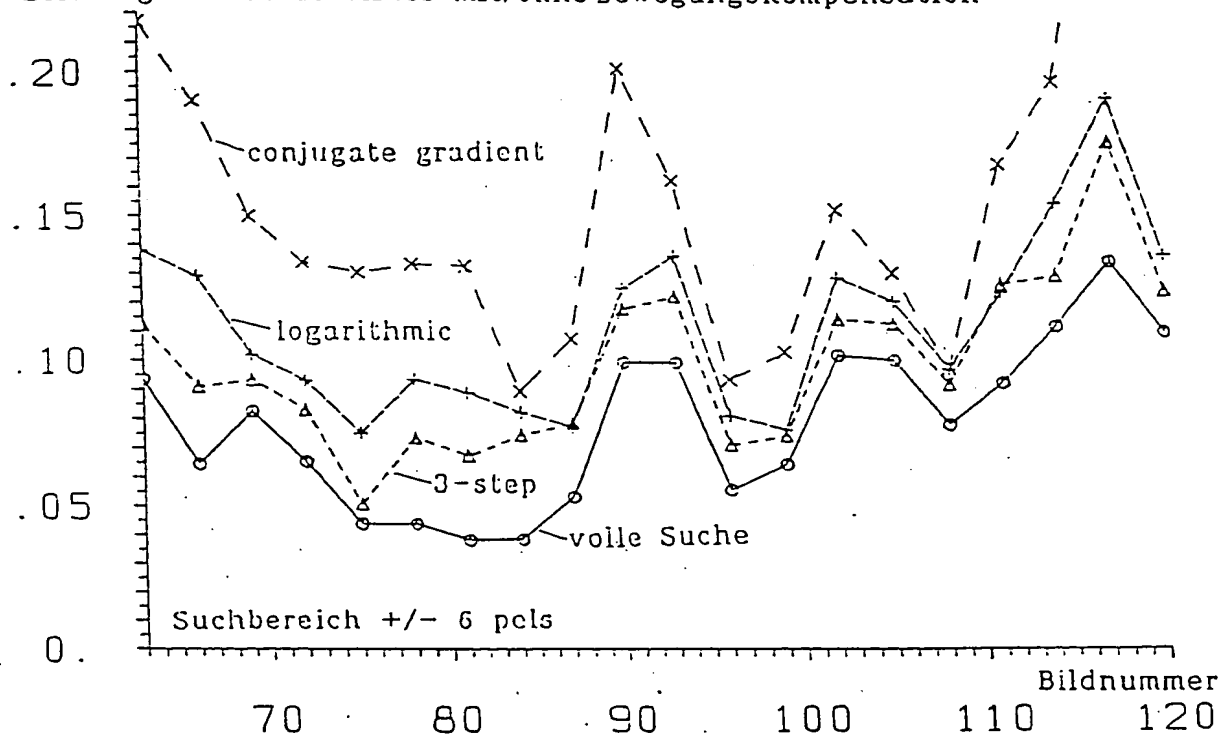


Fig. 9

Abbildung 3.2.6

Leistung des Fehlerbildes mit/ohne Bewegungskompensation

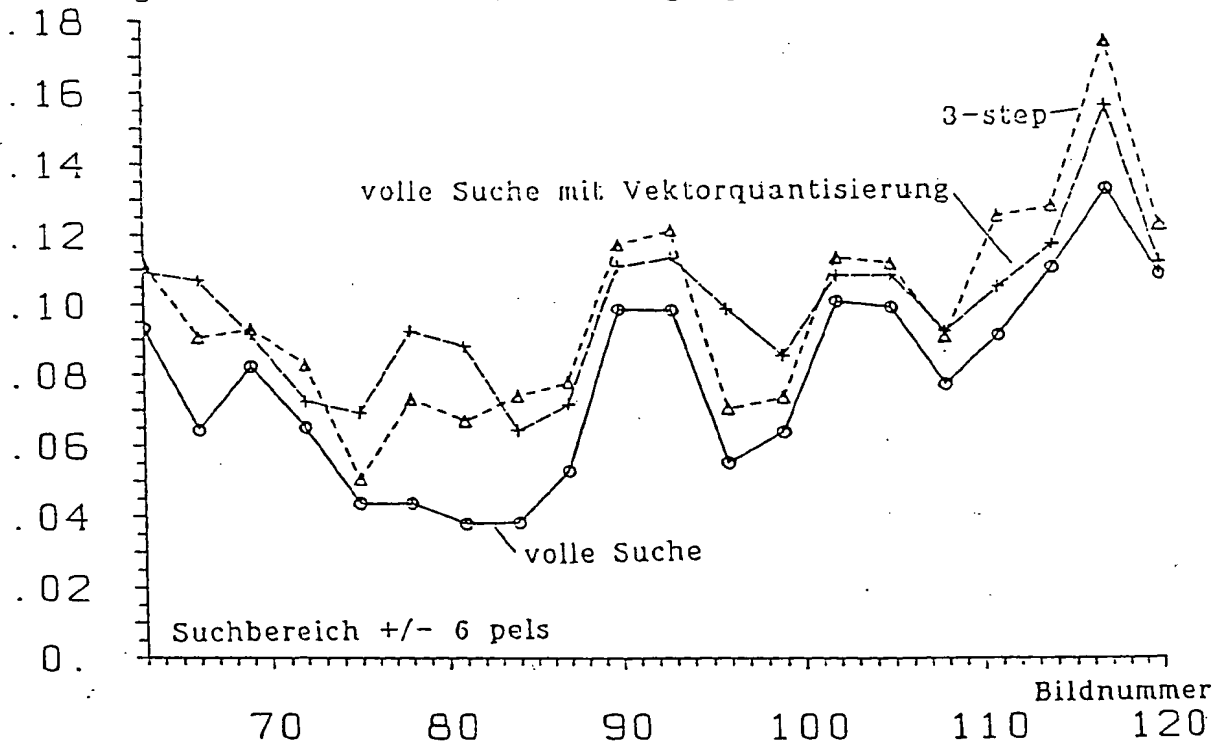


Fig. 10

Abbildung 3.2.7

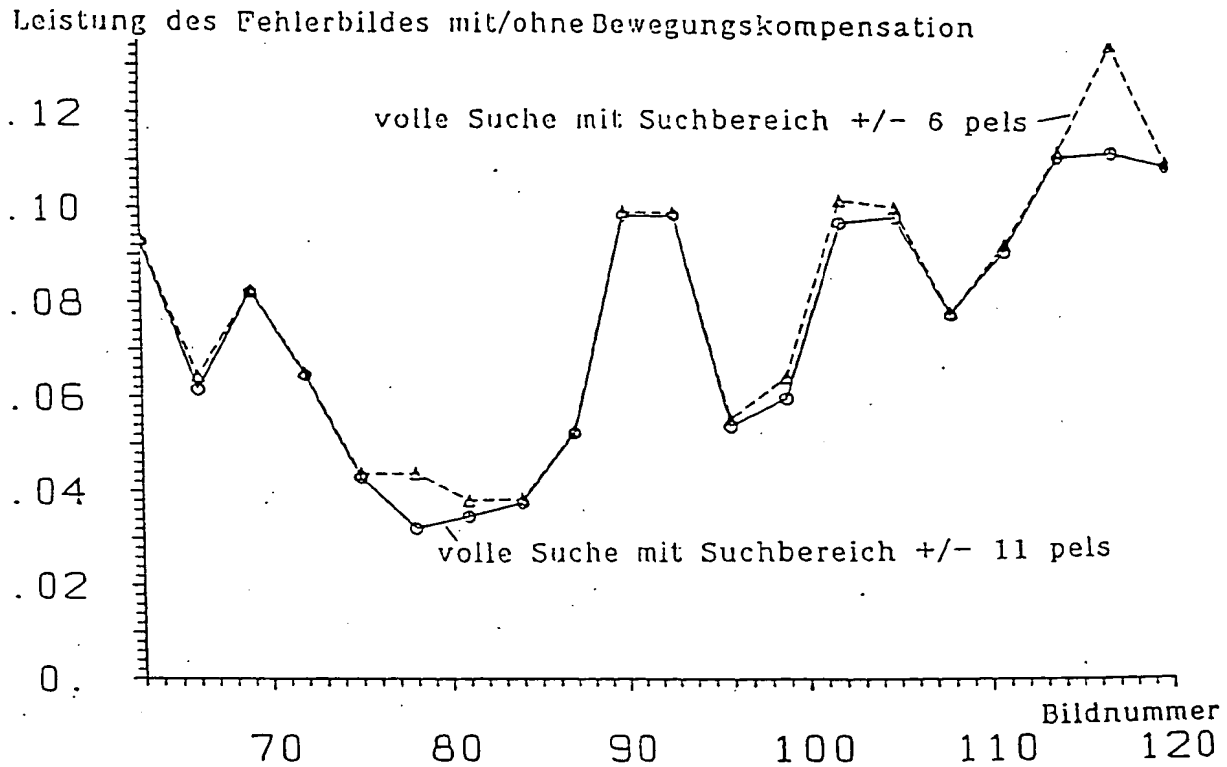


Fig. 11  
Abbildung 3.2.8

# Hierarchische Suche

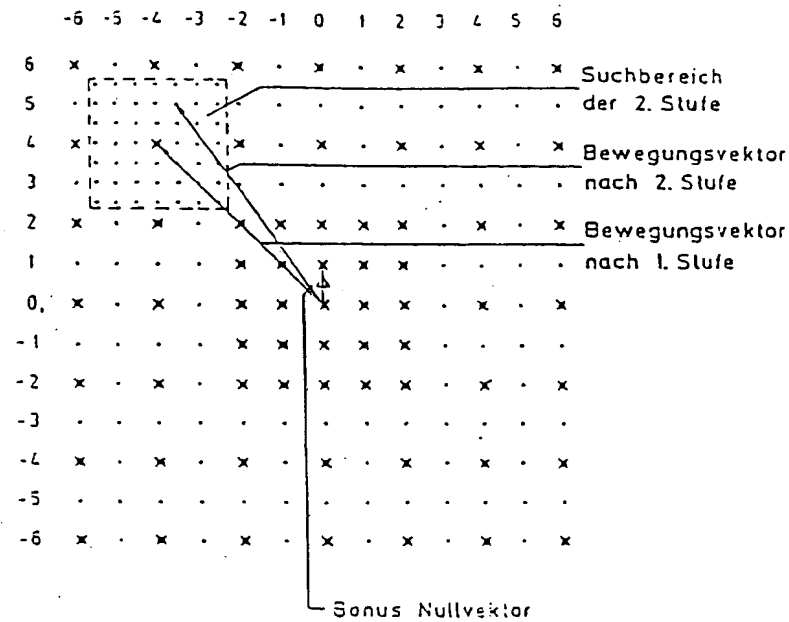


Fig. 12

Abbildung 3.2.9:

## Leistung des Fehlerbildes mit/ohne Bewegungskompensation

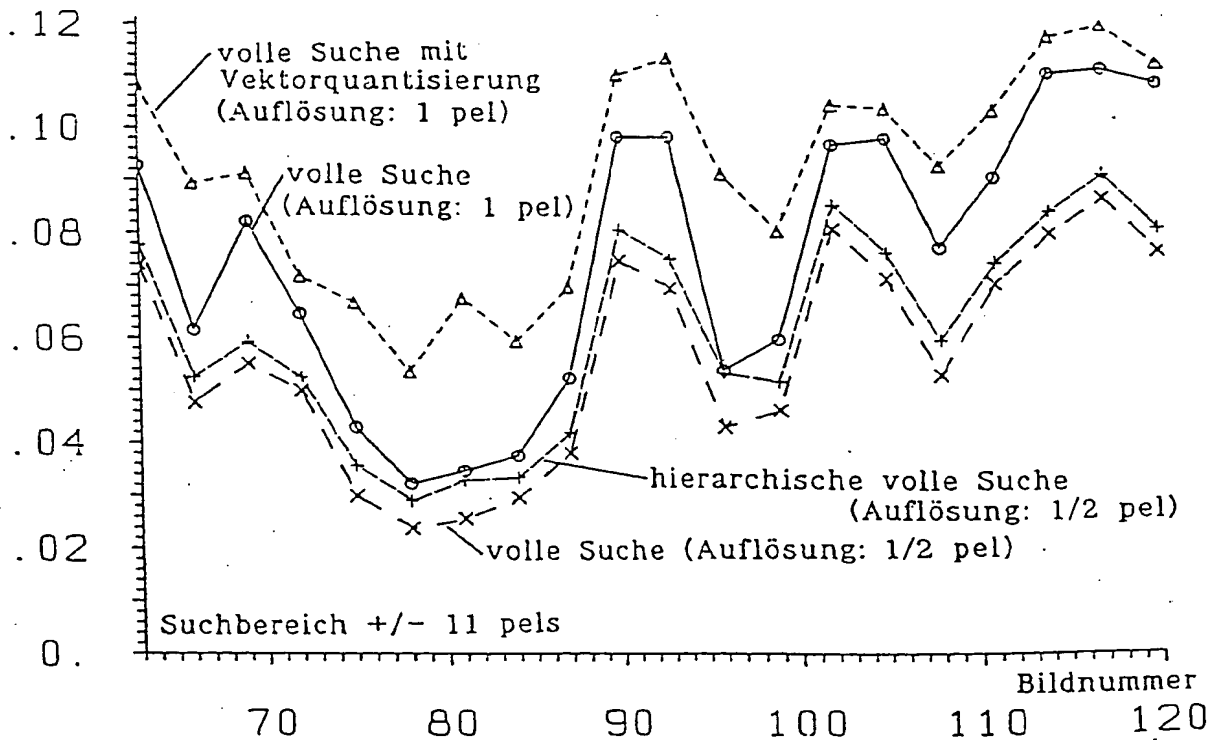


Abbildung 3.2.10:

Fig. 13

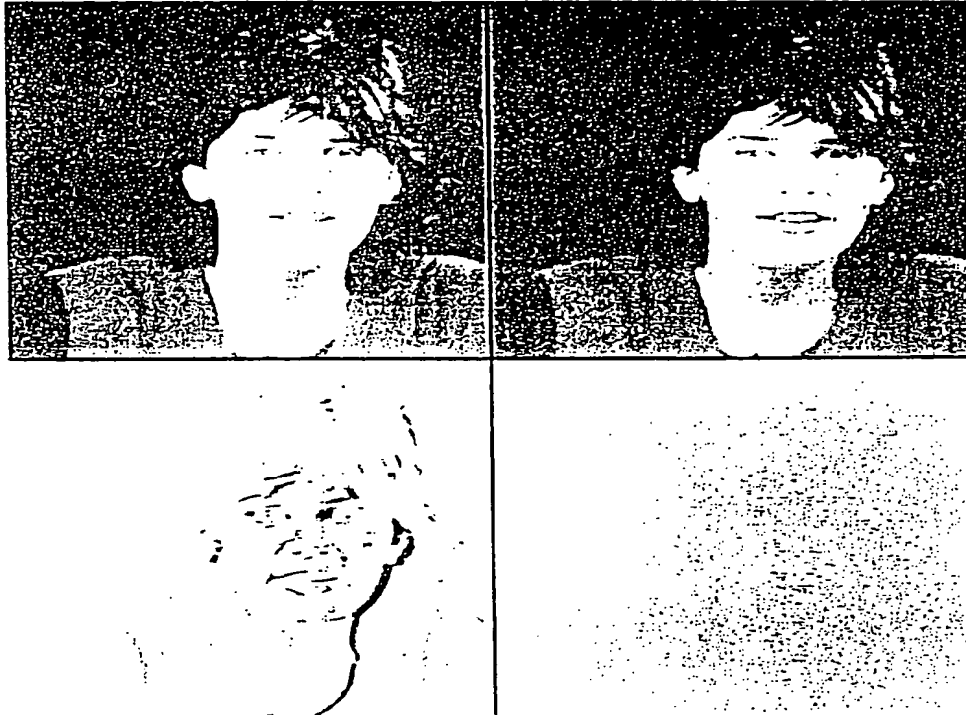


Fig. 14

Abbildung 3.2.11

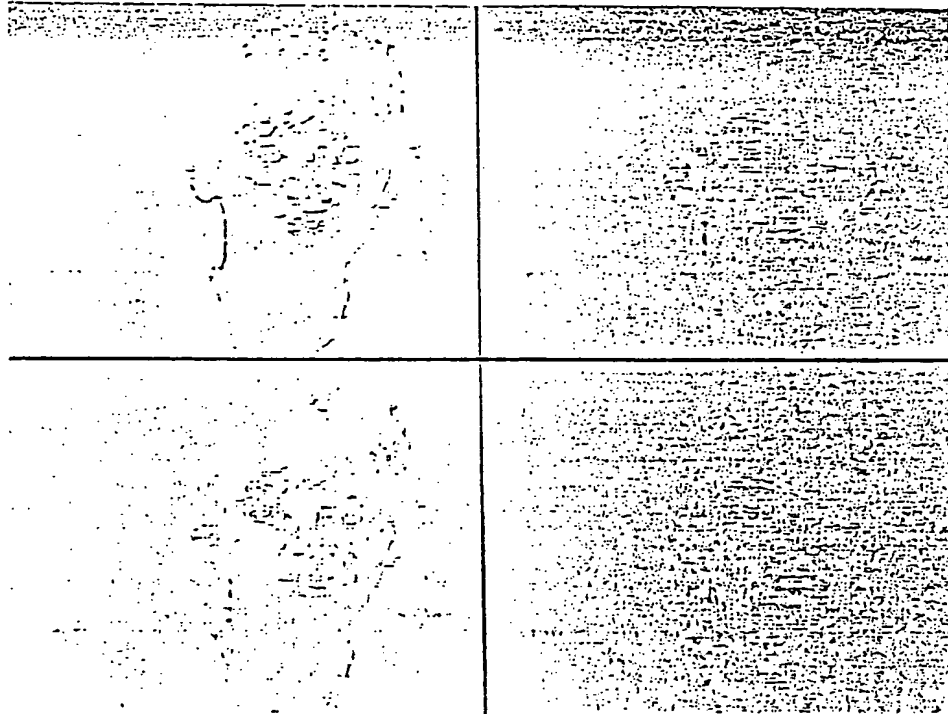


Fig. 15  
Abbildung 3.2.12:



	Komponente	ALEXIS	SPLITSCREEN	TREVOR
Nachbarvektoren in vertikaler und horizontaler Richtung	X	0.31	0.48	0.46
	Y	0.23	0.58	0.46
Nachbarvektoren in diagonalen Richtung	X	0.26	0.35	0.38
	Y	0.16	0.48	0.38
Nachbarvektoren in zeitlicher Richtung	X	0.28	0.38	0.26
	Y	0.18	0.48	0.26

Fig. 16  
Tabelle 3.3.1

	ALEXIS	SPLITSCREEN	TREVOR
Zweidimensionale Entropiecodierung	0.98	4.53	4.53
Interframe Prädiktion	0.98	4.27	5.04
Intraframe Prädiktion	1.14	4.65	4.45

Tabelle 3.3.2

Fig. 17

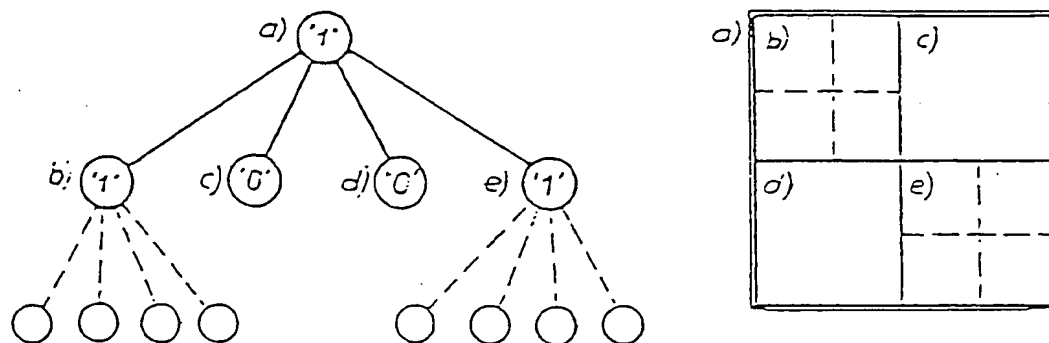


Abbildung 3.3.3

Fig. 18

## \* Zweidimensionale Entropiecodierung

Vektor-Entropie	3.52 bit/Vektor
Gesamtdatenmenge	1394 bit/Bild

## \* Quadtree-Codierung

Vektor-Entropie ohne Nullvektoren	4.45 bit/Vektor
Anteil der Vektoren $\neq (0,0)$	49,8 %
Ortsinformation, Quadtree	241 bit/Bild
Code für Vektoren	878 bit/Bild
Gesamtdatenmenge	1119 bit/Bild

Tabelle 3.3.4

Fig. 19

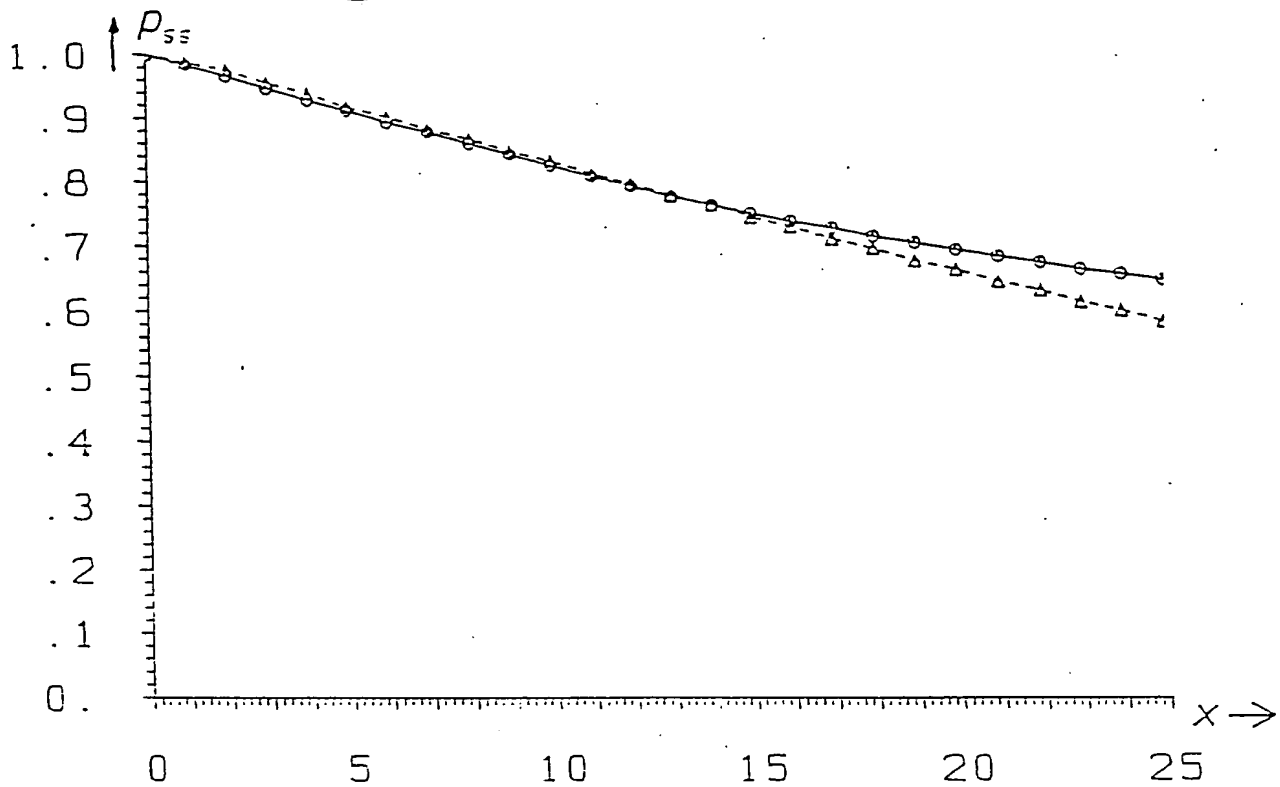


Abbildung 4.1.1

Fig. 20

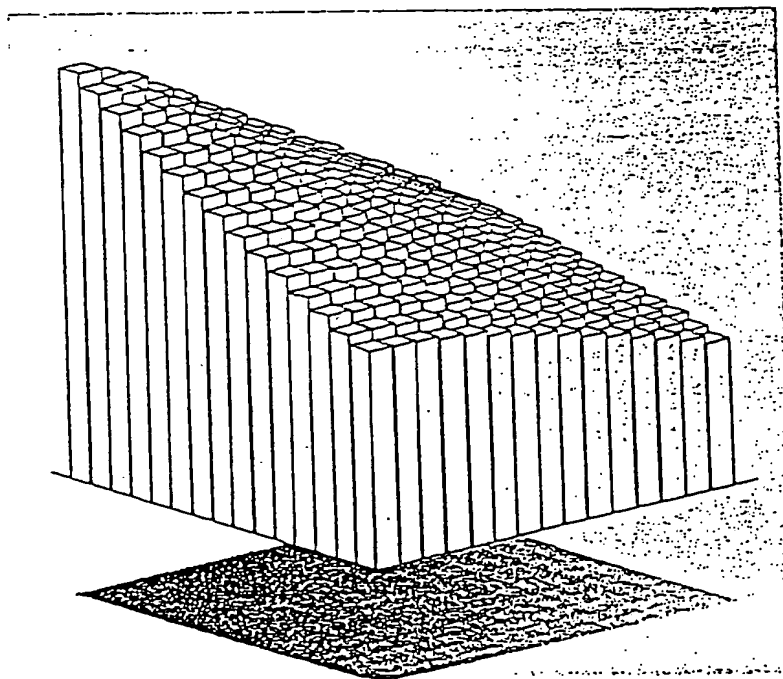


Abbildung 4.1.2

Fig. 21

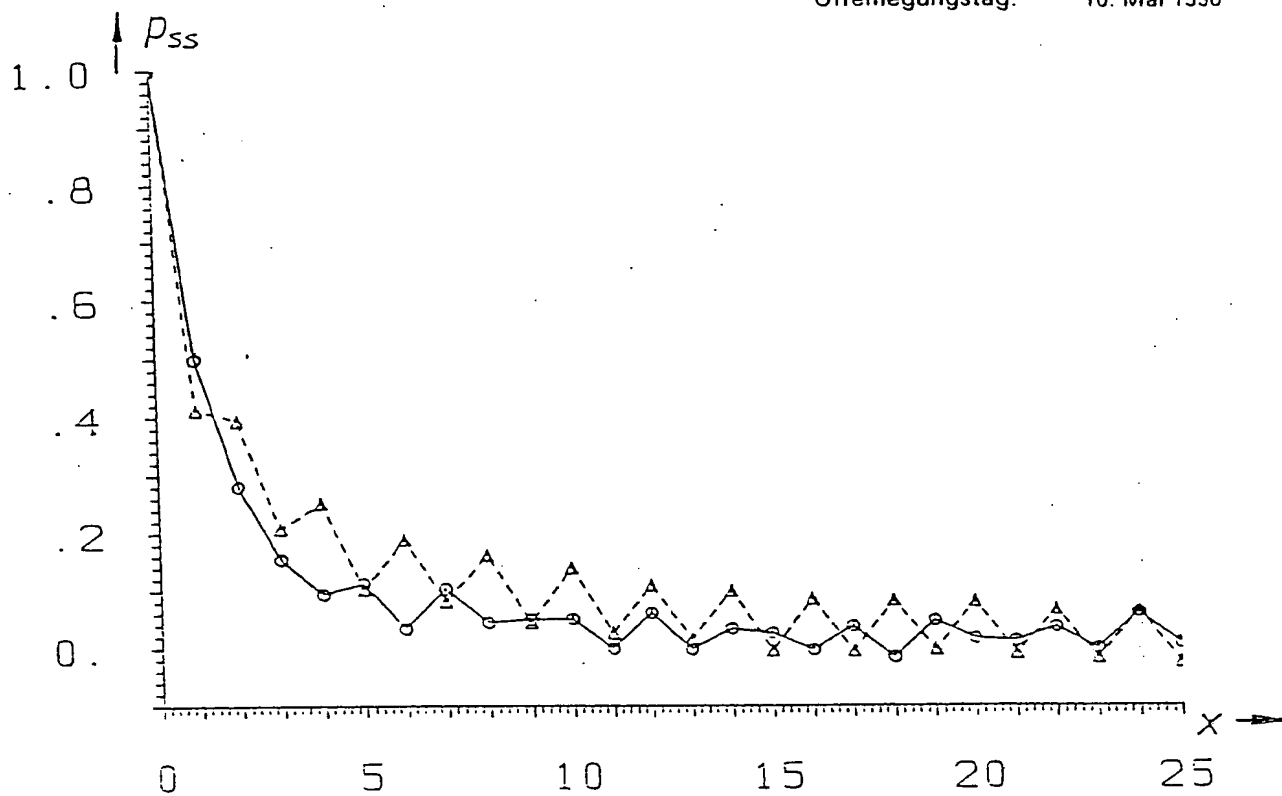


Abbildung 4.1.1

Fig. 22

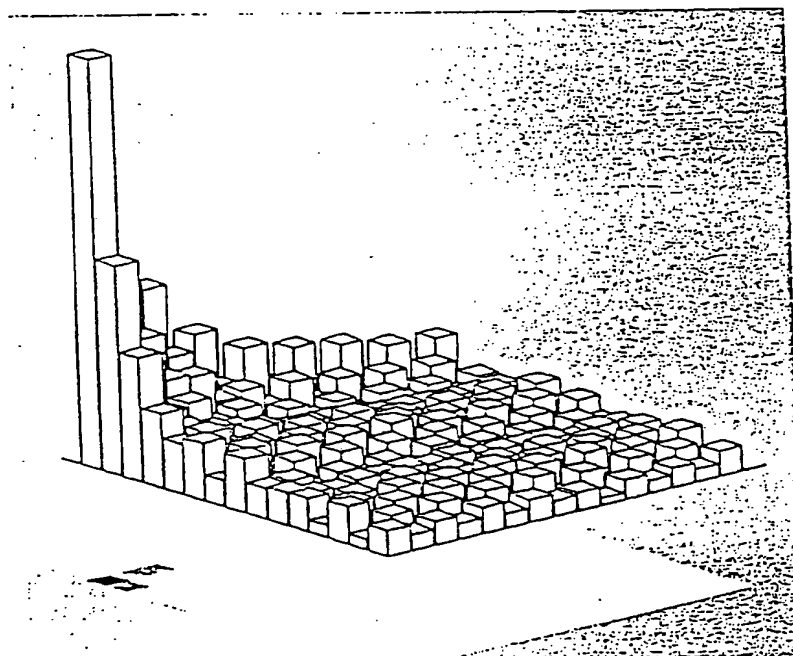


Abbildung 4.1.4

Fig. 23

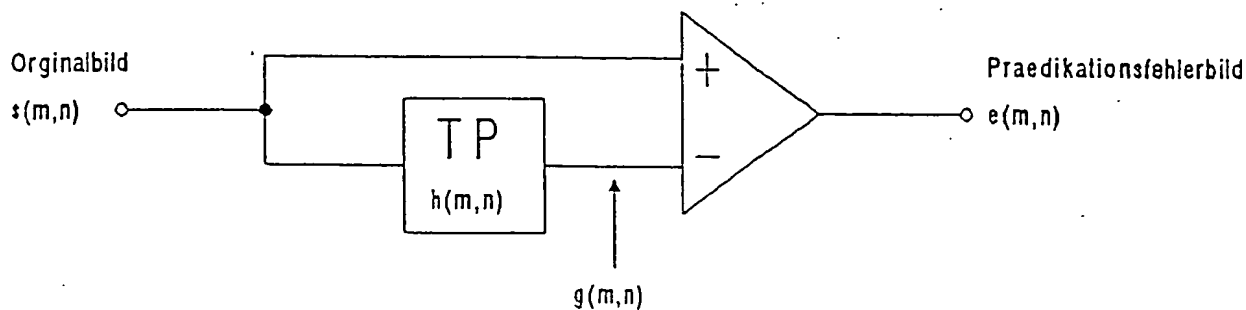


Fig. 24  
Abbildung 4.1.5

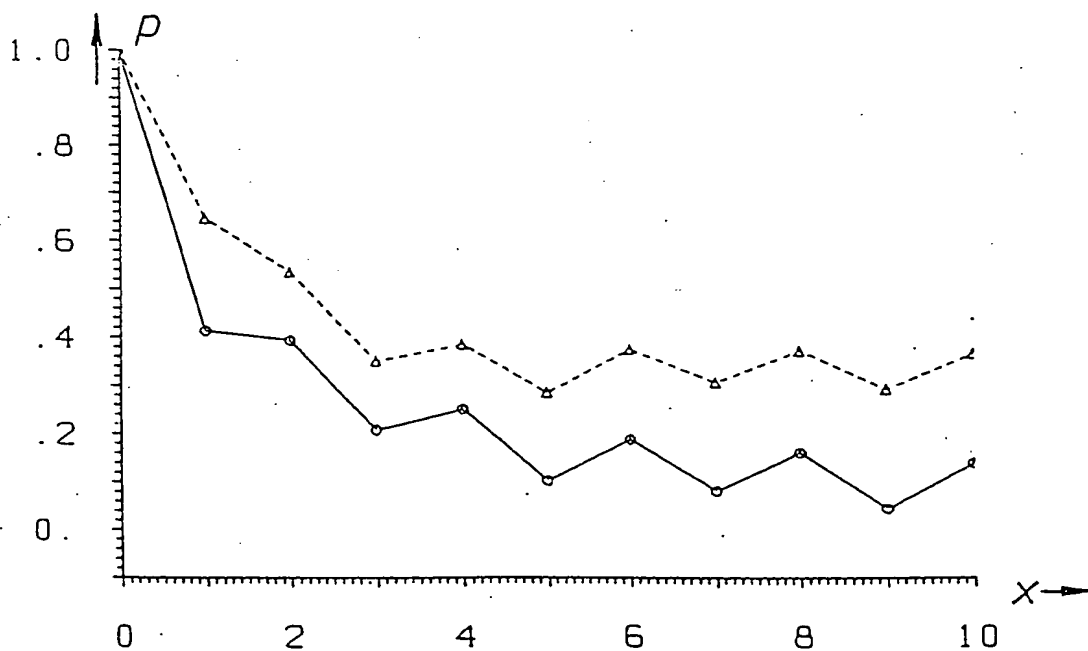


Abbildung 4.1.6

Fig. 25

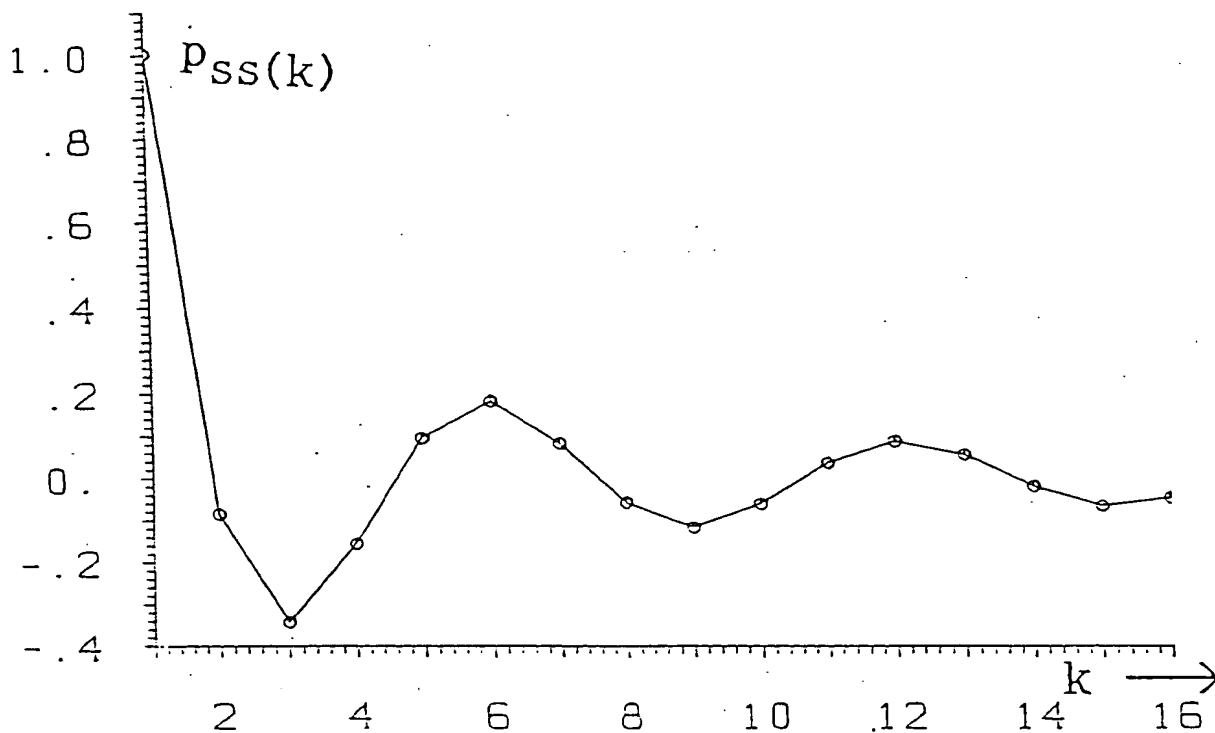


Abbildung 4.1.7

Fig. 26

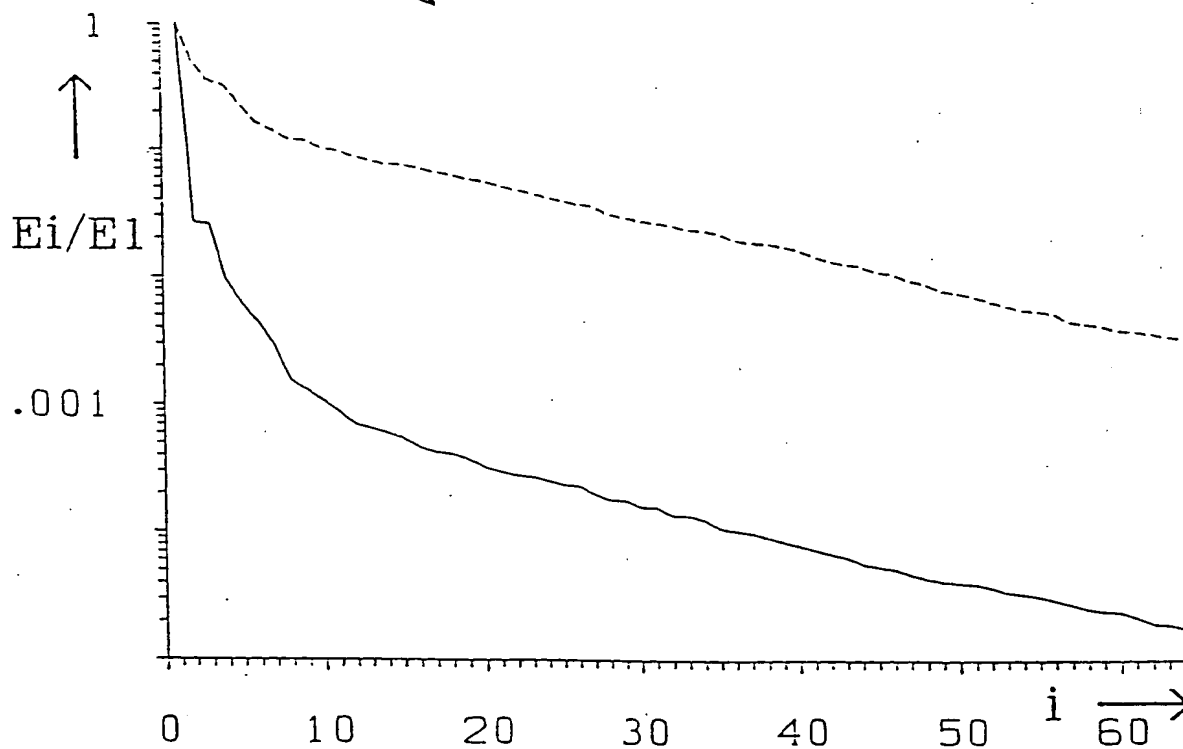


Abbildung 4.2.1

Fig. 27

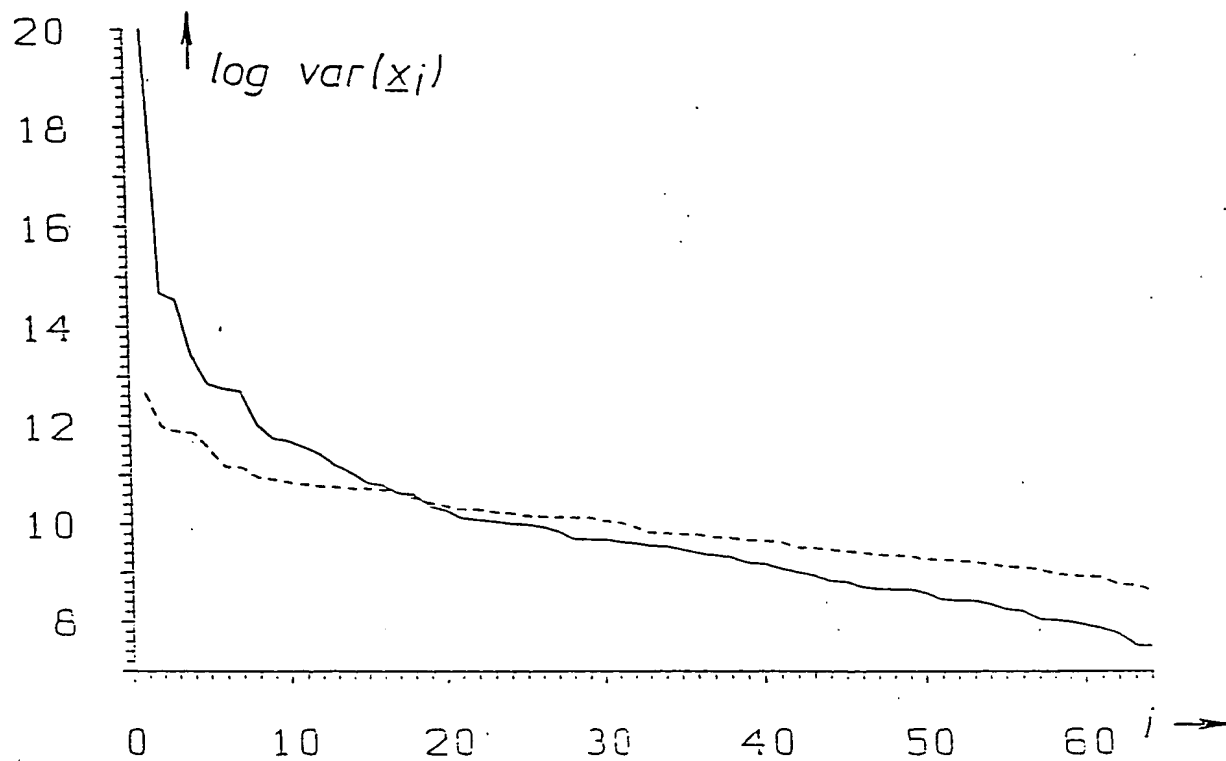


Abbildung 4.2.2

Fig. 28

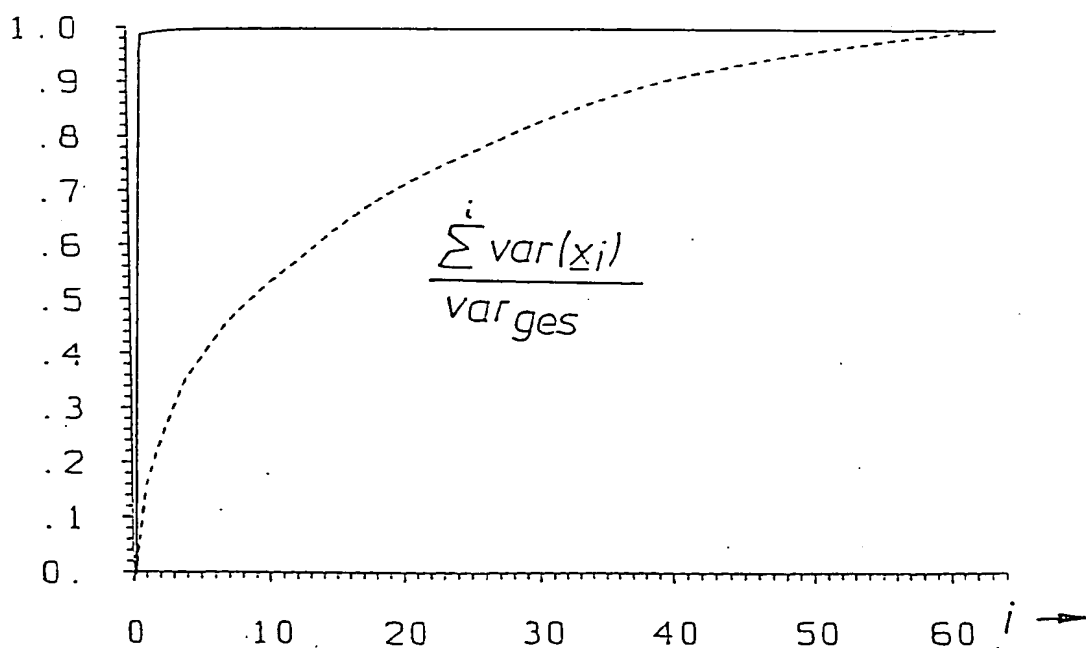


Abbildung 4.2.3:

Fig. 29

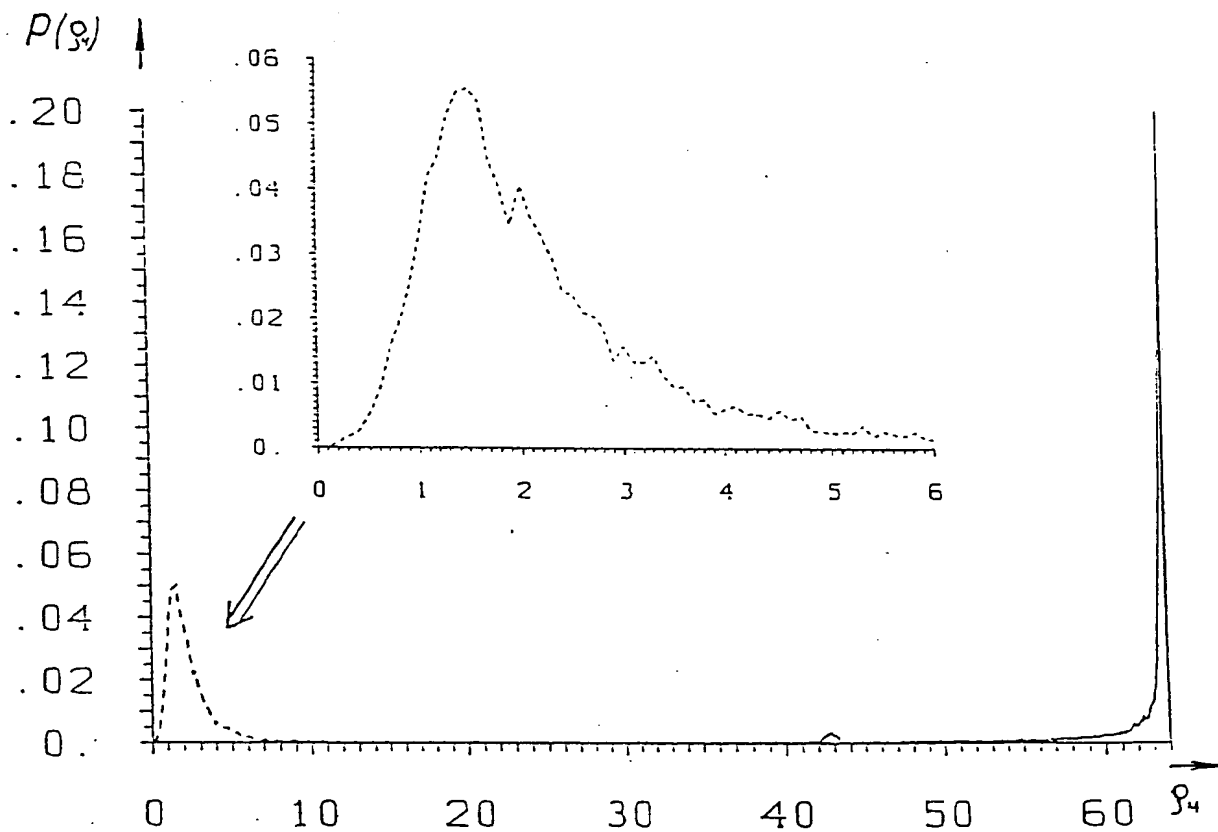


Abbildung 4.2.4

Fig. 30



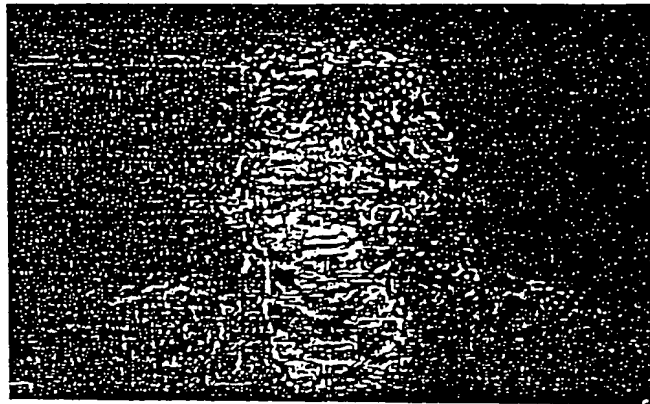


Abbildung 4.3.1 Fig. 31

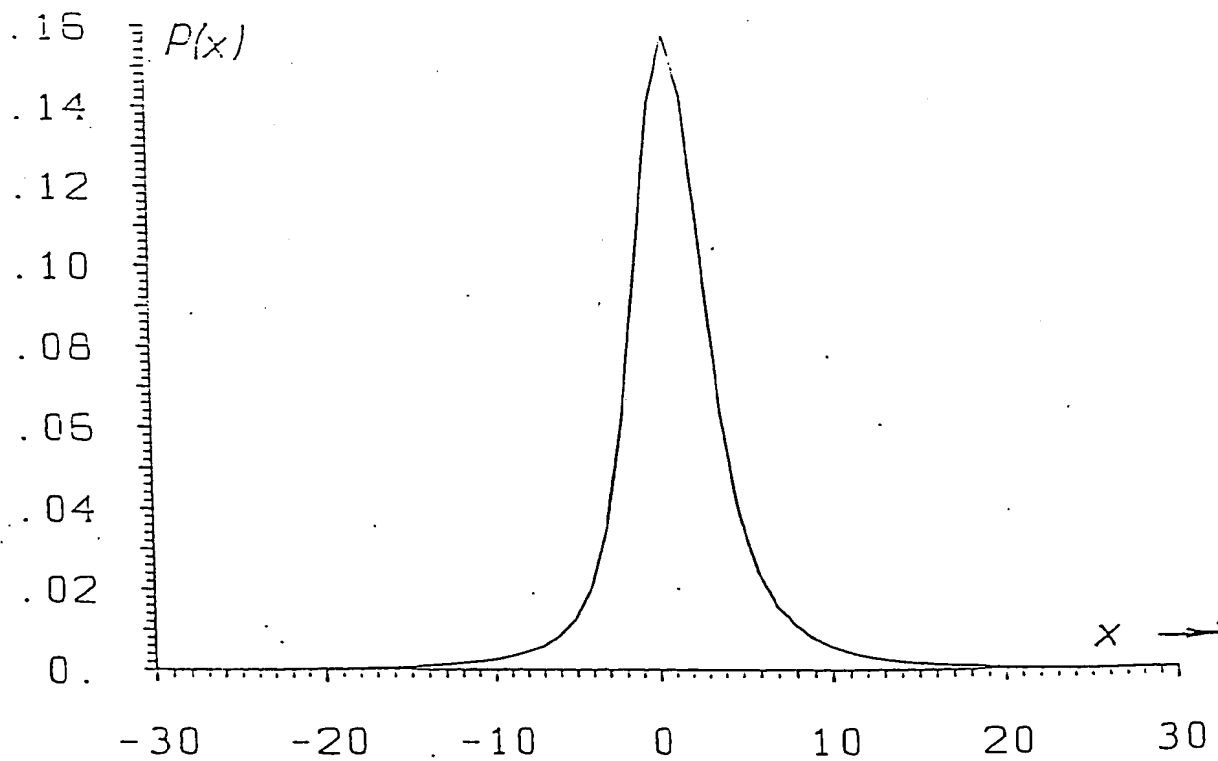


Abbildung 4.3.2 Fig. 32

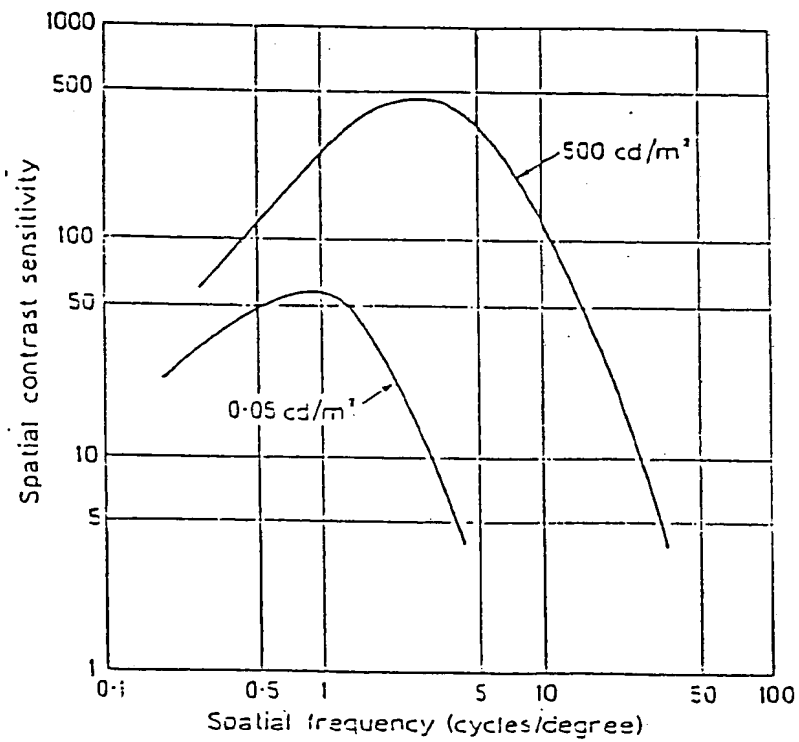


Abbildung 4.3.3

Fig. 33

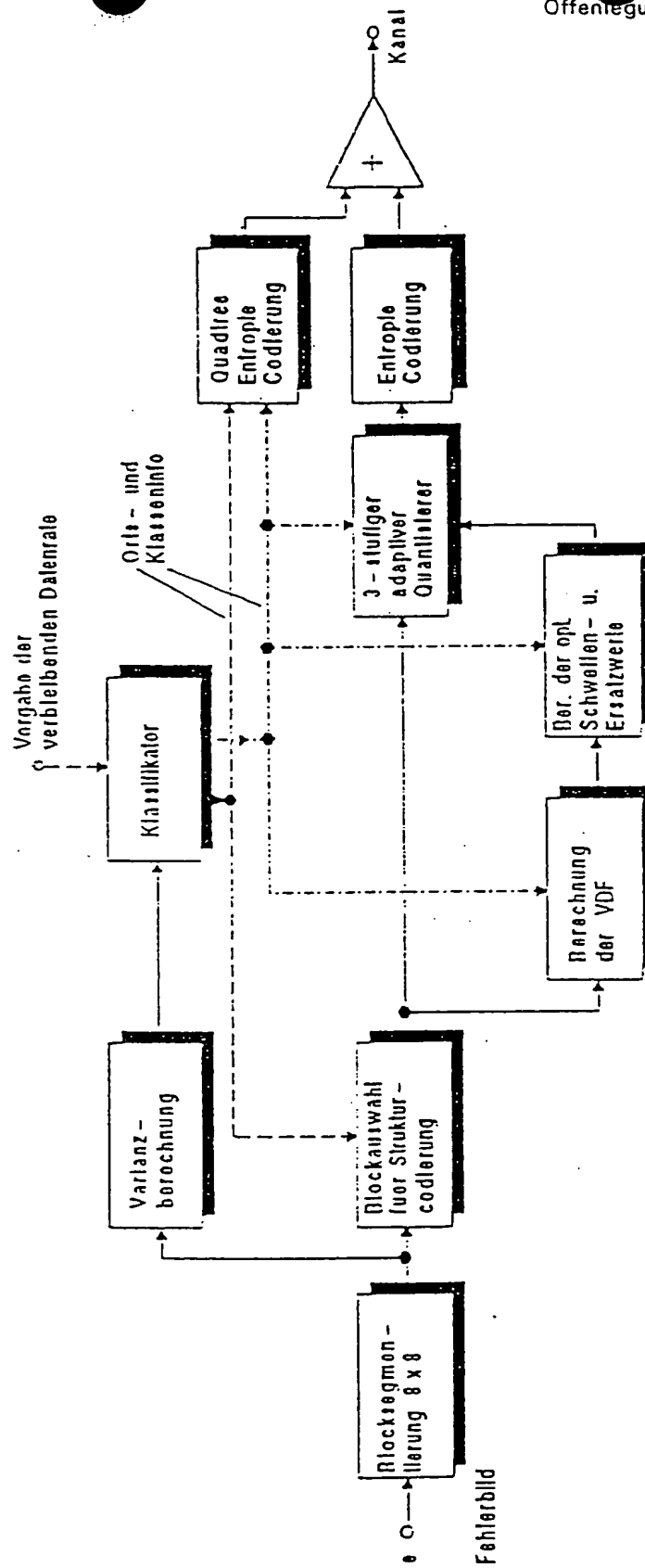


Fig. 34

Abbildung 4.3.4

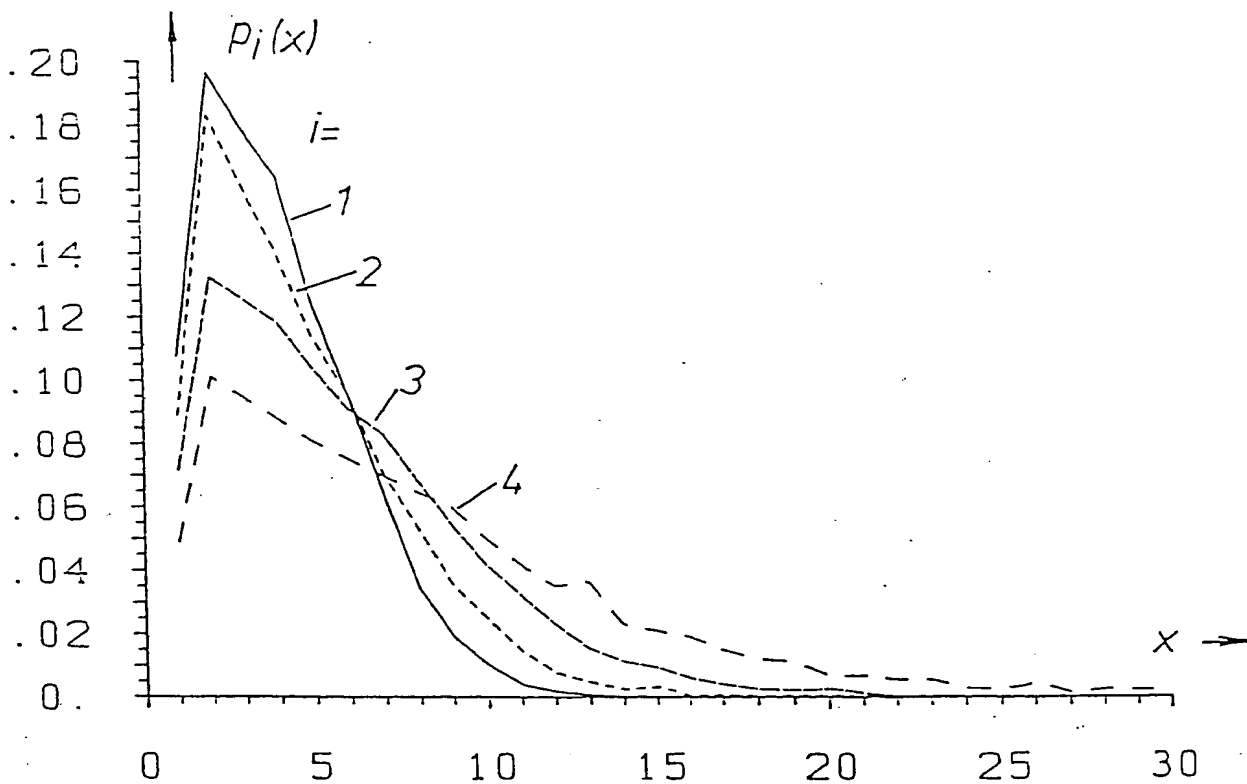


Abbildung 4.3.5

Fig. 35

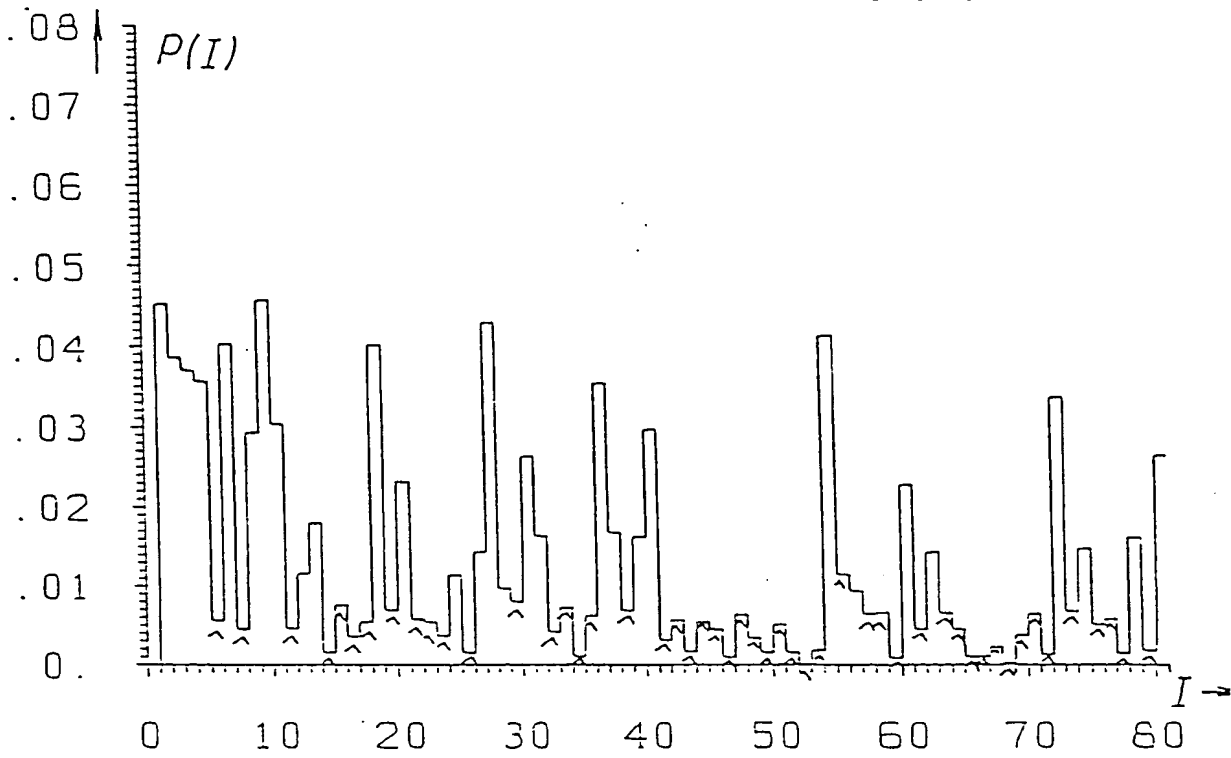


Abbildung 4.3.6 Fig. 36

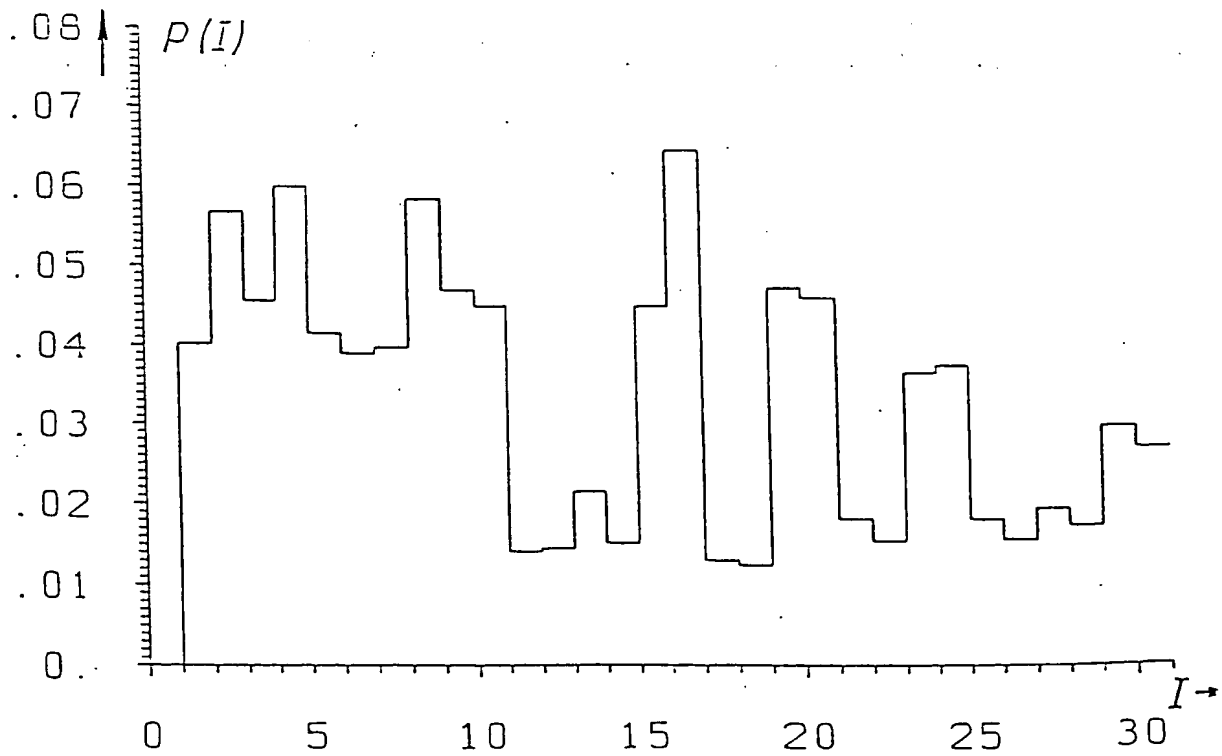


Abbildung 4.3.7 Fig. 37

SENDER

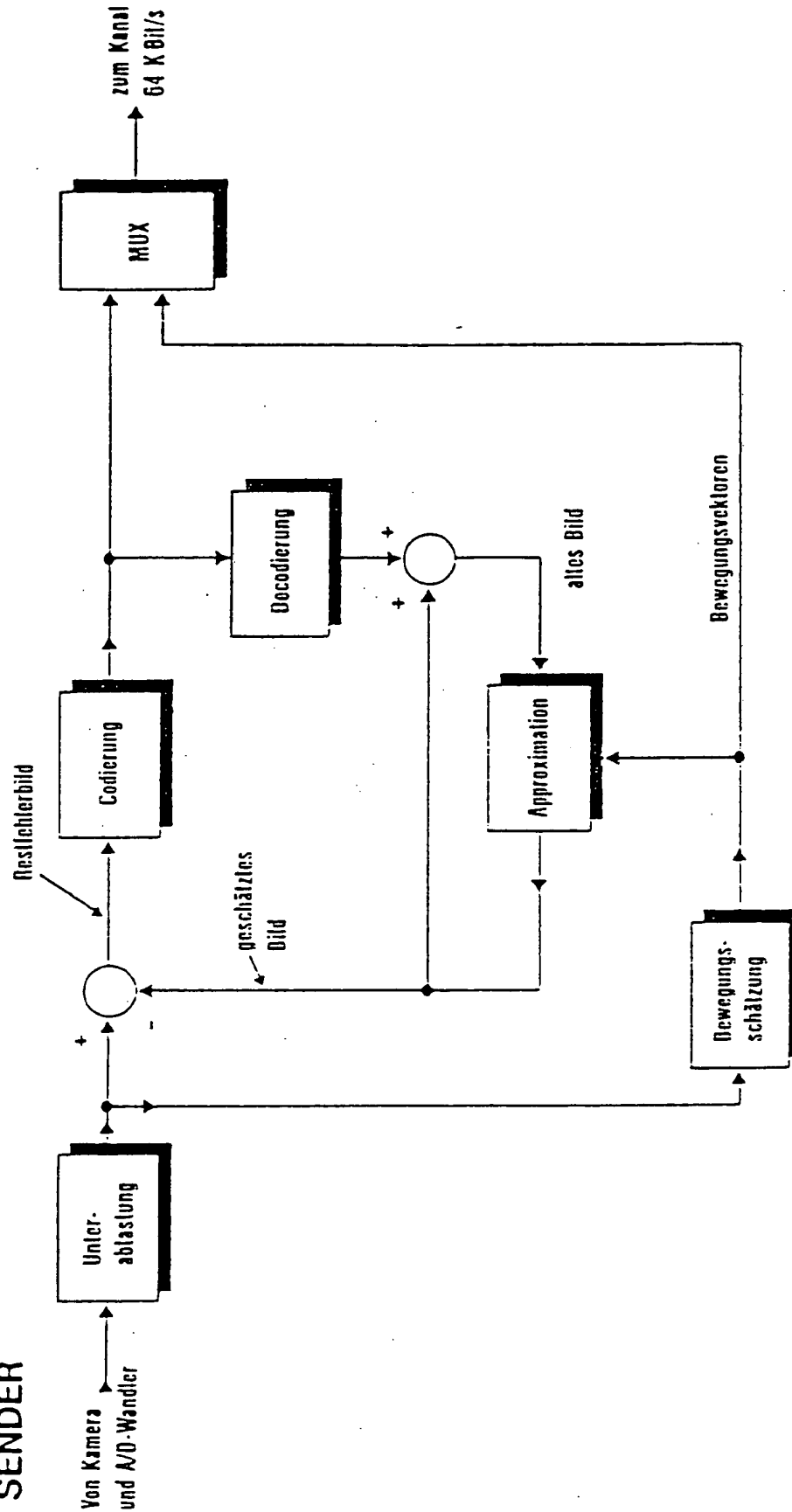


Fig. 38

Abbildung 4.4.1a

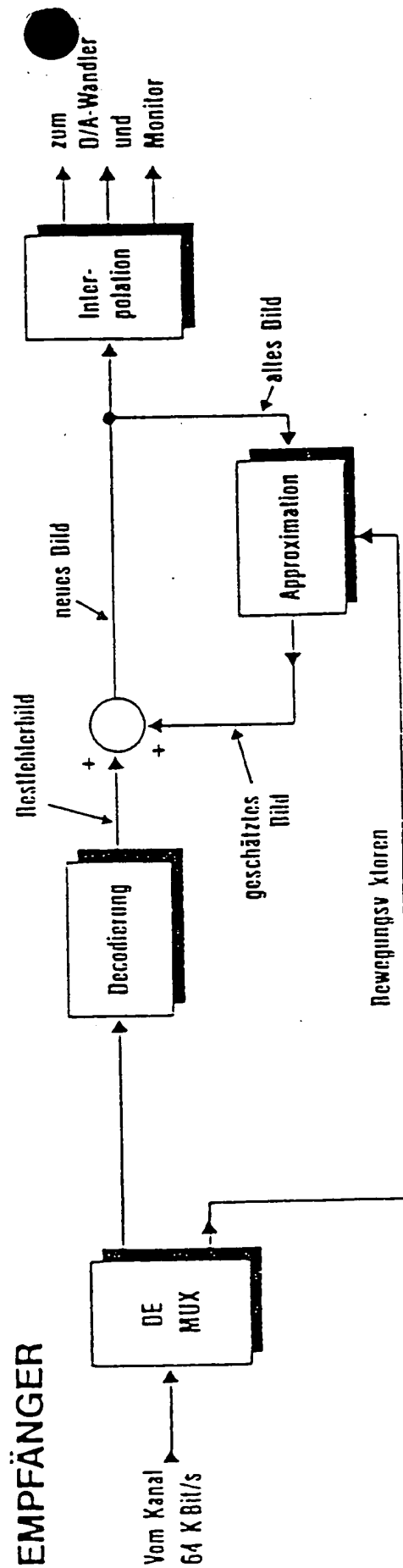


Fig. 39

Abbildung 4.4.1.a

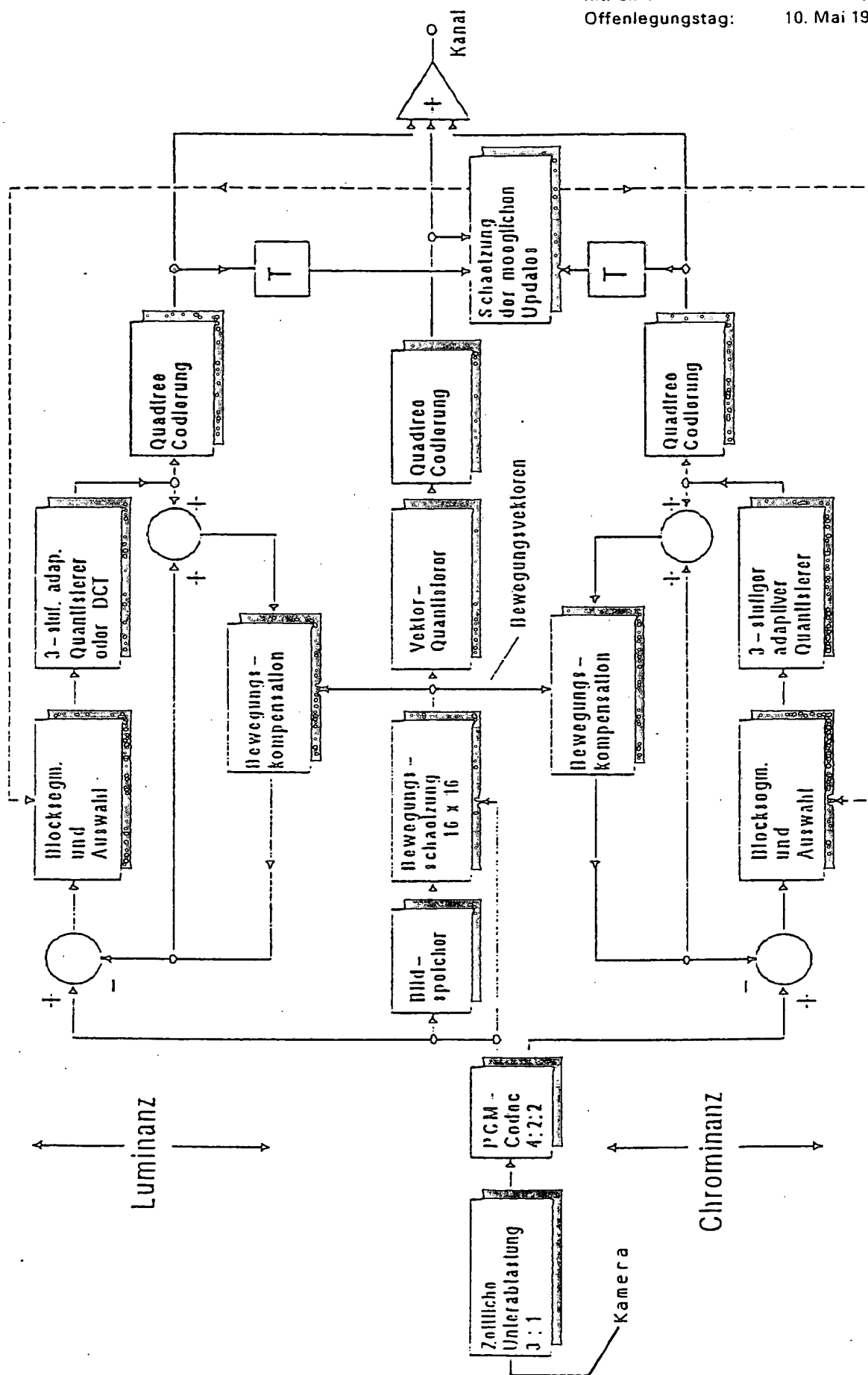


Abbildung 4.4.1.b

Fr. 40







Fig. 43

Abbildung 4.4.4